eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
BILBOKO INGENIARITZA ESKOLA
ESCUELA DE INGENIERÍA DE BILBAO

MÁSTER UNIVERSITARIO EN

INGENIERÍA DE TELECOMUNICACIÓN

TRABAJO FIN DE MÁSTER

ADECUACIÓN DE UNA BASE DE DATOS BILINGÜE EN EUSKERA Y CASTELLANO PARA SU USO EN CONVERSIÓN DE TEXTO EN HABLA

Estudiante *Buruchaga Ramos, Aingeru*
Directora *Navas Cordón, Eva*
Departamento **Ingeniería de Comunicaciones**
Curso académico *2020 - 2021*

Bilbao, 28 de febrero de 2021

Resumen

En el desarrollo de sistemas de conversión de texto en habla, para que la voz artificial generada tenga buena calidad, es fundamental que la voz del locutor con la que se genera tenga también buena calidad. Por ello, estos sistemas se desarrollan normalmente usando bases de datos cuidadosamente diseñadas y grabadas para este propósito. Sin embargo, esta es una tarea muy costosa, tanto desde el punto de vista económico como por el tiempo que requiere. Si no se graba una base de datos expresamente para desarrollar el sistema de conversión de texto en habla, es necesario adaptar grabaciones ya existentes para su uso en conversión de texto en habla. Disponer de ficheros de audio que cumplan las condiciones necesarias no es sencillo, bien porque en los ficheros disponibles la voz del locutor no tiene buena entonación e inteligibilidad o porque existe más de un locutor, y, por tanto, voces de diferentes personas. En este segundo caso, separar la voz del locutor de interés es imprescindible para poder emplear los ficheros.

Una manera de separar los diferentes locutores que hablan en un espacio de tiempo es mediante sistemas de diarización de locutores. Este tipo de sistemas permiten diferenciar cuántos locutores hablan en un fichero de audio, determinando en qué instantes habla cada uno de ellos.

En este trabajo se estudia la adecuación de diferentes sistemas de diarización de locutores a una base de datos que contiene voces de periodistas de EITB junto con voces de otros locutores. Se realiza la implementación y optimización con dos sistemas de diarización y se comprueba su adecuación a la base de datos por medio de los resultados obtenidos. De esta manera, se consigue desarrollar una herramienta que permite obtener ficheros de voz válidos para desarrollar sistemas de conversión de texto en habla.

Laburpena

Testua hizketa bihurtzeko sistemak garatzean, funtsezkoa da sortzen den esatariaren ahotsa kalitate ona izatea, sortutako ahots artifizialak kalitate ona izan dezan. Horregatik, sistema horiek, kontu handiz diseinatutako eta horretarako grabatutako datu-baseak erabiliz garatzen dira. Hala ere, lan hori oso garestia da, bai ekonomiaren aldetik, bai behar duen denboragatik. Testua hizketa bihurtzeko sistema garatzeko datu-base bat grabatzen ez bada, grabazioak egokitu behar dira. Ez da erraza beharrezko baldintzak betetzen dituzten audio-fitxategiak edukitzea, bai eskuragarri dauden fitxategietan esatariaren ahotsak intonazio eta ulergarritasun onik ez dutelako, bai esatari bat baino gehiago dagoelako, eta, beraz, hainbat pertsonen ahotsak daudelako. Bigarren kasu horretan, ezinbestekoa da esatari nagusiaren ahotsa bereiztea fitxategiak erabili ahal izateko.

Denbora tarte batean hitz egiten duten esatariak bereizteko modu bat da diarizazio sistemak erabiltzea. Sistema mota horiei esker, audio-fitxategi batean zenbat esatari hitz egin duten bereiz daiteke, bakoitzak zein unetan hitz egiten duen zehaztuz.

Lan honetan, diarizatzeko sistemak EITBko kazetarien eta beste esatari batzuen ahotsak dituen datu-base batera egokitzapena aztertzen da. Inplementazioa eta optimizazioa bi diarizazio-sistemaren bidez egiten da, eta lortutako emaitzen bidez datu-basera egokitzen direla egiaztatzen da. Horrela, testua hizketa bihurtzeko sistemak garatzeko balio duten ahots-fitxategiak lortzeko tresna garatzea lortzen da.

Abstract

In order to obtain high quality artificial voices, text to speech systems require good quality input audio recordings. Therefore, these systems are normally developed using carefully designed and recorded databases. However, this is a very expensive task, both economically and due to the time it takes to develop them. If a database is not recorded specifically to develop text to speech conversion systems, it is necessary to adapt existing recordings to the requirements of these systems. Obtaining those recordings is not an easy task, due to poor conditions in the speaker's voice or because the main speaker´s voice is mixed with others in the recording. For that reason, it is very important to separate the different voices in the recordings.

Speaker diarization systems detect speaker changes in a recording and they can be used to detect when a speaker talks. In this work, speaker diarization systems are studied with an EITB (*Basque Radio and Television*) database, in which journalists and other speakers are present. The main objective is to set up two diarization systems and check if they are suitable for the database of this work. After that, an effective speaker diarization tool is provided to get recordings for text to speech purposes.

Lista de siglas

AHC: Agglomerative Hierarchical Clustering

BD: Base de datos

BIC: Bayesian Information Criterion

DER: Diarization Error Rate

EITB: Euskal Irrati Telebista

GMM: Gaussian Mixture Model

KBM: Key Background Model

MFCC: Mel Frequency Cepstral Coefficients

NIST: National Institute of Standards and Technology

RTTM: Rich Transcription Time Marked

TTS: Text to Speech

UBM: Universal Background Model

VAD: Voice Activity Detection

Índice

Lista de figuras

Lista de tablas

1. Introducción

El uso de tecnologías relacionadas con el procesamiento de la voz ha aumentado en los últimos años, empleándose en una gran cantidad de aplicaciones, como en el reconocimiento de voz, verificación e identificación de locutor y traducción automática entre diferentes lenguajes.

Otra aplicación en la que el desarrollo de las tecnologías del habla ha permitido un avance importante es en los sistemas que convierten de texto a habla, conocidos como **TTS** (*Text to Speech*). Estos sistemas producen habla sintética y para su desarrollo es necesario contar con grabaciones de voz junto con su texto correspondiente. Por ello, aspectos que influyen en la calidad de la grabación como una buena entonación e inteligibilidad del locutor y la falta de cortes en la señal de audio determinarán que el sistema TTS final sea válido o no y su rendimiento.

La calidad y cantidad de los datos de partida en la creación de un sistema TTS son determinantes para la calidad de voz sintética obtenida. Obtener suficientes grabaciones de calidad no es tarea fácil, y menos en idiomas minoritarios como el euskera. Por esta razón, es importante aprovechar todo el contenido que se disponga. En algunos casos, las grabaciones contienen voz de diferentes locutores mezcladas, pero para el desarrollo de un sistema TTS se debe utilizar la voz de un único locutor o en el caso de un sistema TTS multilocutor, se deben marcar de quién es cada voz.

La diarización de locutores o ***speaker diarization*** permite determinar los intervalos de tiempo en los que habla cada locutor en una grabación de audio, identificando los diferentes locutores en una grabación y los instantes en los que habla cada uno de ellos. Mediante esta técnica, es posible separar los diferentes locutores que intervienen en una misma grabación para posteriormente extraer la parte correspondiente a un único locutor y emplearla en el desarrollo de un sistema TTS.

En este trabajo, se ha estudiado y adecuado una base de datos proporcionada por EITB (*Euskal Irrati Telebista*) para utilizarla en un sistema TTS. Esta base de datos contiene audios de periodistas junto con sus transcripciones en texto, cumpliendo con las condiciones de calidad de los audios iniciales, además de disponer de material tanto en euskera como en castellano.

Sin embargo, estos ficheros contienen partes de audio que no corresponden al periodista, sino que son fragmentos correspondientes a otros locutores relacionados con la noticia. Por ello, es imprescindible el estudio, implementación y evaluación de sistemas de diarización de locutores para conseguir únicamente la parte del audio correspondiente al periodista y poder desarrollar en un futuro un sistema TTS con esa información.

En la Figura 1 se muestra el procedimiento seguido en este trabajo. Inicialmente, se dispone de una base de datos de EITB con ficheros que contienen audio tanto del periodista que narra la noticia como de personas entrevistadas. Tras un estudio de la base de datos, se hace una selección de ficheros sobre los que se hará un procesamiento inicial para utilizarlos en los sistemas de diarización. Este procesamiento inicial consistirá principalmente en eliminar los silencios existentes en los ficheros, que dan lugar a un aumento del error en la diarización. Posteriormente, los métodos de diarización estudiados e implementados, se utilizan con la

selección de ficheros. Se llevan a cabo diferentes configuraciones en los sistemas, para comprobar con cuál de ellas se obtienen mejores resultados.

Una etapa posterior a la obtención de los resultados, y que es imprescindible en este trabajo es la de identificación del locutor principal, que, en este caso, corresponde al periodista. De esta forma, se hará la evaluación de los sistemas de diarización no sólo de manera general (teniendo en cuenta los errores de todos los locutores existentes) sino que, además, se obtendrá el error cometido específicamente en la parte del periodista, que tiene mayor interés en este trabajo.

Finalmente, en base a los resultados obtenidos, se podrá comprobar la adecuación de los sistemas de diarización empleados con la base de datos (BD). De esta manera, se puede determinar la validez de esos sistemas de diarización para poder aplicarlos a grabaciones de audio de diferentes locutores y poder utilizarlas para conseguir ficheros que permitan desarrollar sistemas TTS.

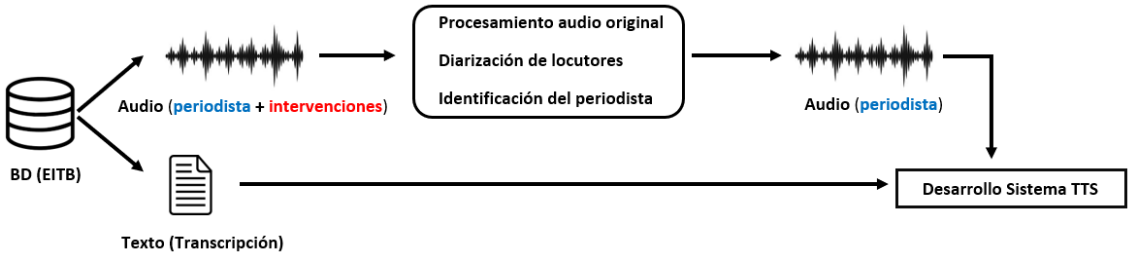


Figura 1: Esquema para la adecuación de la BD para el desarrollo de un sistema TTS

2. Contexto

La evolución tecnológica producida durante los últimos años, unida a los avances en investigación, han permitido un gran desarrollo en muchos ámbitos. Uno de los campos donde ha habido impacto positivo ha sido en las tecnologías del habla, que cada vez están más presentes en la sociedad.

Entre las técnicas desarrolladas gracias al avance en las tecnologías del habla se encuentran las siguientes: reconocimiento automático del habla, verificación/identificación del locutor, traducción automática, identificación del idioma, y síntesis del habla, que es la generación del habla de manera artificial. La síntesis del habla ha permitido utilizar la voz en el desarrollo de sistemas TTS, que permiten la conversión de texto a voz.

Un sistema TTS convierte un texto de entrada en una voz sintética en un idioma determinado, con el objetivo de que la voz resultante tenga buenas condiciones de inteligibilidad, naturalidad y comprensibilidad. La arquitectura de un sistema TTS se puede ver en la Figura 2, donde se puede apreciar que el proceso se divide en una parte de procesamiento de texto y en los métodos de generación del habla.

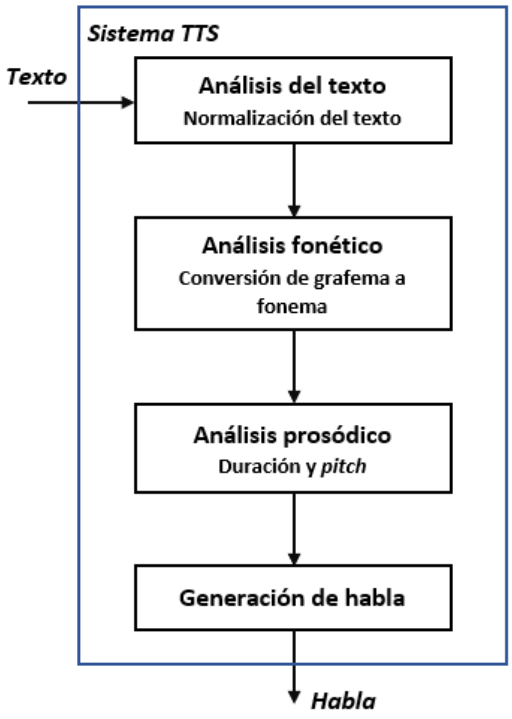


Figura 2: Arquitectura de un sistema TTS. Imagen adaptada de [1]

- *Procesamiento de texto:* Esta etapa consiste en tres partes: una primera de normalización de palabras no estándar, la conversión de grafema a fonema y un análisis prosódico que permita predecir la duración y entonación adecuada.
- *Métodos de generación del habla:* Permiten la conversión de la secuencia fonética a una señal de voz. Ejemplos de estos métodos son la síntesis paramétrica, la síntesis concatenativa y la síntesis por formantes, que ofrecen diferentes resultados en función de

la inteligibilidad y naturalidad de la voz resultante y el coste computacional y la cantidad de información necesaria para su implementación [2].

En el desarrollo de un TTS dos factores son fundamentales para que el resultado final sea una voz de calidad. Por un lado, cuanto mayor sea la cantidad y la calidad de material disponible, mejor será el resultado, debido a que se habrá desarrollado con más palabras y frases diferentes. Por otro lado, que las grabaciones utilizadas contengan únicamente voz de un único locutor es imprescindible para que el TTS se implemente con la voz del locutor de las grabaciones.

En cuanto al contenido, cada vez es mayor la cantidad de material disponible gracias a la evolución de los métodos de adquisición y almacenamiento de todo tipo de contenido audiovisual. Además, ese aumento de contenido ha permitido que se pueda tener señal de voz no sólo de medios tradicionales como pueden ser llamadas telefónicas o entrevistas de radio, sino que las señales de voz pueden provenir de entornos muy diferentes como de noticias de televisión o vídeos de Internet, lo que permite que haya mucha más variedad en los audios de entrada para el desarrollo de tecnologías del habla.

A pesar de que se disponga de mayor contenido, es importante distinguir qué parte de ese contenido es útil para poder desarrollar aplicaciones que trabajen con señales de voz. En general, se debe identificar la parte del audio que contenga las voces de los locutores y eliminar la parte que no sea de interés pero que se encuentra en el audio debido a las características del entorno donde se ha grabado, como pueden ser aplausos si la grabación corresponde a un programa de televisión o tráfico de fondo si se ha grabado en el exterior.

En el caso de un sistema TTS, se debe extraer, además de lo comentado anteriormente, las voces de los locutores que no pertenezcan al locutor principal de quién se quiere desarrollar la voz resultante. La diarización de locutores, que se explicará en detalle en el siguiente apartado, consiste en identificar los diferentes locutores que hablan en una misma grabación y determinar los instantes en los que habla cada uno de ellos. Esta técnica se puede resumir en dar respuesta a la pregunta *"¿quién ha hablado y cuándo?"* y como resultado se obtiene un identificador del locutor que ha hablado en cada segmento en los que se divide el audio.

Las primeras investigaciones en este campo comenzaron a finales de los 90, aunque no fue hasta el año 2000 cuando se adoptó el término de diarización de locutores en las evaluaciones de *Rich Transcription (RT)* promovidas por el NIST. Durante los años posteriores, gracias tanto a las evaluaciones del NIST [1], que promueven y supervisan los avances en tecnologías de reconocimiento automático del habla, como a ETAPE[2] (*Évaluations en Traitement Automatique de la Parole*), REPERE (*Reconnaissance de PERsonnes dans des Emissions audiovisuelles*) o la más reciente DIHARD[3], ha sido posible evaluar los diferentes sistemas de diarización con una gran cantidad de *datasets* y comprobar su rendimiento en base a los resultados obtenidos.

[1] Página web de la evaluación del NIST: https://www.nist.gov/itl/iad/mig/rich-transcription-evaluation
[2] Página web de la evaluación *ETAPE: https://www.aclweb.org/anthology/L14-1022/*
[3] Página web de la evaluación DIHARD: *https://dihardchallenge.github.io/dihard1/overview.html*

3. Estado del arte

En este apartado se introduce la diarización de locutores, explicando, en primer lugar, en qué consiste y sus características. Después, se describen los diferentes bloques que componen el proceso de diarización y las diferentes técnicas que se pueden emplear en cada uno de ellos. Finalmente, se analiza la evaluación de los resultados que se obtienen tras el proceso de diarización.

3.1 Sistemas de diarización de locutores

En este apartado se explica el funcionamiento de un sistema de diarización de locutores, que permita identificar los instantes en los que habla cada persona en una señal de audio. Es la parte de mayor importancia del trabajo, debido a que cuanto mayor sea la precisión en la identificación de los instantes que corresponden a cada locutor en los audios, será menor el error en el audio final que sólo contenga la parte hablada por el periodista.

La diarización de locutores consiste en dividir en segmentos un audio de entrada y asociarlos al locutor correspondiente, identificando los instantes de comienzo y final de cada locutor. A modo de resumen, la diarización responde a la pregunta "¿quién ha hablado y cuándo?" en un entorno en el que hay varios locutores [3]. En la Figura 3 se puede ver el resultado de la diarización en una señal de audio en la que hablan tres locutores diferentes:

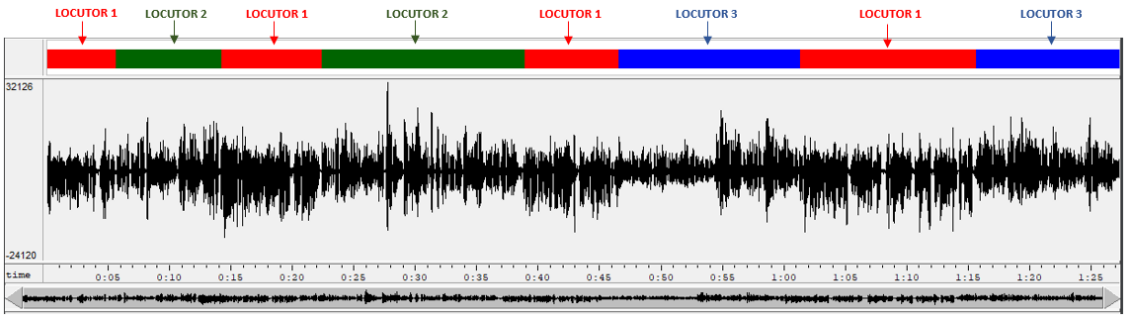


Figura 3: Diarización de señal de audio con 3 locutores

En general, un sistema de diarización de locutores está formado por las siguientes componentes:

- **Parametrización** *(features extraction)***:** Es la extracción de las características que representan la señal de voz, como pueden ser los coeficientes MFCC. Se extraen para cada segmento en los que se ha dividido el audio de entrada.
- **Detección de actividad vocal:** Detección de los segmentos que corresponden a voz, eliminando el resto de segmentos.
- **Segmentación** (*speaker segmentation*): Identificar los instantes en los que se produce un cambio de locutor en el audio de entrada
- **Agrupamiento** (*clustering*): Determinar qué segmentos de la señal de voz corresponden al mismo locutor, en función de la información extraída en la parametrización y obtener el número de locutores existentes en el audio.

- **Resegmentación:** Es una etapa opcional que consiste en utilizar la información extraída para hacer una nueva segmentación y ajustar los límites entre los locutores.

La Figura 4 ilustra perfectamente el esquema general de un sistema de diarización.

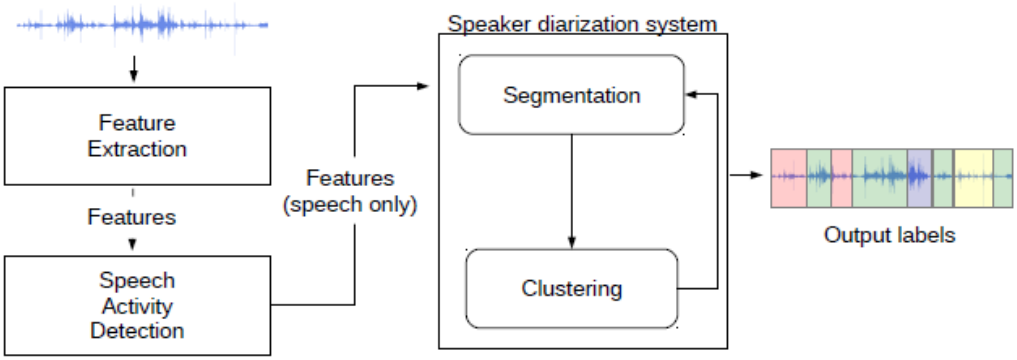


Figura 4: Esquema general de un sistema de diarización de locutores [4]

Los procesos de segmentación y agrupamiento pueden ser implementados de manera independiente o ser considerados como un mismo bloque en un proceso iterativo en el que el resultado del agrupamiento se usa de entrada en la segmentación para optimizar los resultados en cada etapa.

Los sistemas que realizan el proceso de diarización pueden ser sistemas *offline* u *online*. Los sistemas que hacen el procesamiento *offline* disponen de toda la información a procesar previamente a realizar el proceso. Por otro lado, los sistemas *online* sólo disponen de la información grabada hasta ese momento, lo que implica mayor complejidad y retardo en la obtención de los resultados.

Sin embargo, el esquema del proceso de diarización no se restringe únicamente a estas fases, sino que puede haber más en función de la información de entrada y del objetivo que se busque con la diarización. A modo de ejemplo, en el caso de reconocimiento de locutor, es necesaria una etapa adicional que permita distinguir las características de diferentes locutores como el género o la edad, como paso previo a identificar la identidad del locutor.

A continuación, se describe en detalle cada bloque que forma el proceso de diarización completo, explicando con mayor detalle los bloques de parametrización y agrupamiento, que son las etapas que mayor importancia han tenido en la implementación de los sistemas de diarización utilizados en este trabajo.

3.1.1 Parametrización

La parametrización es la extracción de características de la señal de voz del audio de entrada que representen sus propiedades. Los diferentes locutores que contenga el audio tienen características diferentes, que son empleadas para poder identificar si los diferentes segmentos de audio pertenecen al mismo locutor o no.

La señal de voz es una señal de evolución lenta, por lo que se si se examina en intervalos cortos de tiempo (típicamente entre 5 y 100ms), sus características son prácticamente estacionarias

[5]. Por ello, la extracción de características de la señal de voz se suele hacer en ventanas de duración 10 -20 ms, y a menudo se utiliza solapamiento entre ventanas.

El método más más popular para la extracción de características son los *Mel Frequency Cepstral Coefficients (MFCCS)*, aunque también se utilizan otros métodos como Linear *Predictive Coding (LPC)*, o los coeficientes *Linear Frequency Cepstral Coefficients (LFCC)* y *Perceptual Linear Predictors (PLP)*.

MFCC

El método más empleado en los sistemas de diarización son los *Coeficientes Cepstrales en escala de Frecuencia Mel* (MFCC). La principal característica de los MFCC es que da mayor importancia a las frecuencias bajas y menor a las altas, lo que se ajusta a las frecuencias auditivas que son percibidas con mayor sensibilidad por el ser humano (20Hz - 20 kHz).

La mayor influencia de las bajas frecuencias se observa en la escala de Mel mostrada en la Figura 5, que sirve para la conversión de hercios a mels. Una variación de frecuencias en frecuencias bajas, supone una mayor variación en mels que la misma variación en frecuencias altas.

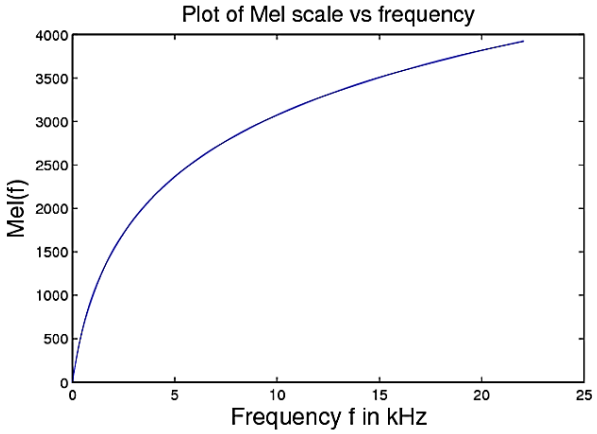


Figura 5: Escala de Mel

La conversión de frecuencia lineal a la frecuencia de Mel se realiza con la siguiente expresión:

$$\mathrm{mel}(f) = 2595 \log\left(1 + \frac{\mathrm{f}}{700}\right)$$

El esquema para la extracción de los MFCC, con los diferentes bloques utilizados en el proceso está representado en la Figura 6.

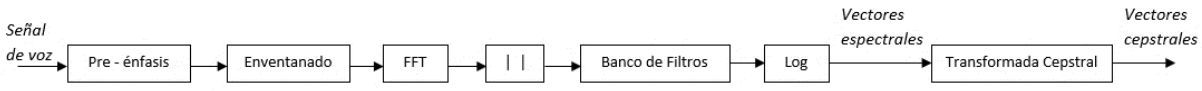


Figura 6: Diagrama de bloques de parametrización mediante MFCC. Figura Adaptada de [6]

- **Pre – énfasis**. Consiste en aplicar un filtro a la señal de voz entrante con el objetivo de mejorar las frecuencias altas del espectro, que normalmente se atenúan en la producción del habla. La señal resultante se obtiene aplicando el siguiente filtro:

$$x_p(t) = \mathrm{x}(t) - \alpha x(t-1)$$

donde α toma valores en el intervalo [0.95, 0.98].

Este filtrado no se aplica siempre, aunque es recomendable. La mejor manera de elegir si aplicarlo o no es comparando los resultados empíricamente.

- **Enventanado**. Si se analiza la señal de voz en intervalos cortos de tiempo (del orden de ms), es posible considerarla como una señal casi-estacionaria. Por ello, el análisis de la señal se hace utilizando ventanas pequeñas, de tamaño mucho menor que la señal completa. Hay dos valores que se deben definir: la longitud de la ventana y el desplazamiento entre dos ventanas consecutivas. Normalmente, los valores que se suelen emplear son tamaños de ventana de 20 – 30ms y un desplazamiento entre ventanas de 10ms.
  Una vez elegidos estos valores, se debe elegir el tipo de ventana a utilizar. Las ventanas comúnmente empleadas son las de tipo *Hamming* y *Hanning*. Estas ventanas suelen ser preferibles que la ventana rectangular, ya que atenúan el efecto de la señal original en los extremos de la ventana actual, reduciendo la influencia de la ventana anterior y posterior y los efectos de borde.
- **FFT (Fast Fourier Transform)**. Se aplica la Transformada Rápida de Fourier de la señal obtenida tras el enventanado. Se debe definir el número de puntos N empleado, generalmente potencia de 2 y que sea mayor que el número de puntos de la ventana. Generalmente se utiliza un valor de N de 512 puntos.
  Posteriormente, se extrae el módulo de la FFT y se obtiene la densidad espectral de potencia de la señal. Como el espectro es simétrico, únicamente se trabaja con la mitad, resultando un espectro de 256 puntos.
- **Banco de filtros.** El espectro obtenido tiene muchas variaciones y para muchos de los objetivos de las tecnologías del habla no se necesitan todos los detalles, únicamente es necesaria su envolvente. Por ello, se suaviza el espectro por medio de un banco de filtros. Un banco de filtros es una serie de filtros paso-banda que se multiplican uno a uno con el espectro para obtener un valor medio en una determinada banda de frecuencia. El banco de filtros se define por su forma (normalmente triangular, aunque puede tener otro tipo de forma) y por la localización de la frecuencia central y el ancho de banda de cada filtro. Para espaciar en frecuencia los filtros, se suelen emplear las escalas de Bark/Mel, adaptada a las frecuencias auditivas que puede escuchar el oído humano. La localización de las frecuencias centrales de los filtros en la escala Mel se establecen por la siguiente fórmula:

$$f_{MEL} = 1000 \cdot \frac{\log(1 + f_{LIN}/1000)}{\log 2}$$

Mientras que las frecuencias centrales de los filtros usando la escala Bark, se obtienen:

$$f_{BARK} = 6 \ln\left(\frac{f_{LIN}}{600} + \sqrt{\frac{f_{LIN}}{600} + 1}\right)$$

En la Figura 7 se muestra una comparativa del banco de filtros Bark y Mel respecto a un banco de filtros con escala lineal. Como se puede observar, los filtros Mel y Bark tiene filtros más estrechos en las frecuencias bajas que en las frecuencias altas, mientras que el banco de filtros en escala lineal, tiene filtros de igual tamaño y espaciados uniformemente en frecuencia. Debido a que el oído humano tiene mayor sensibilidad en bajas frecuencias que en altas frecuencias, estas escalas logarítmicas están más relacionadas con la percepción del sonido que la escala lineal. Para el cálculo de los coeficientes MFCC, se aplica el banco de filtros en escala Mel.

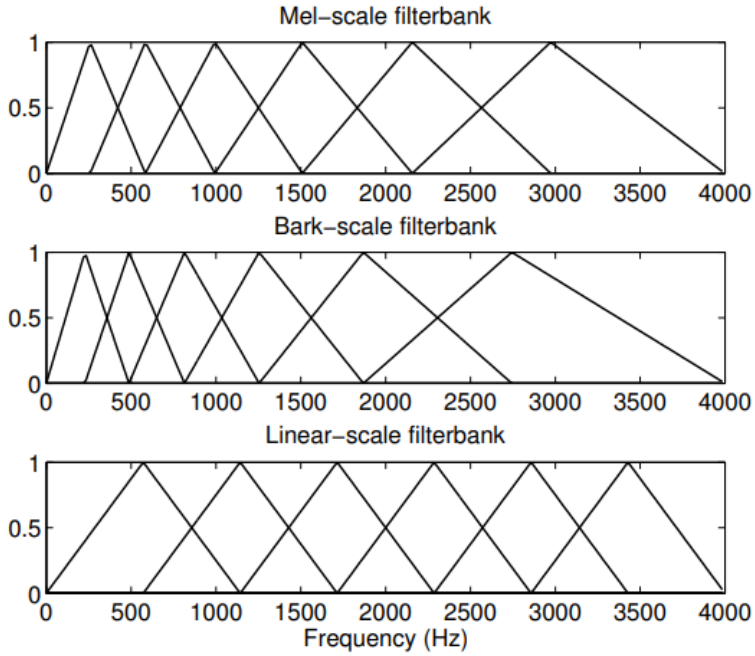


Figura 7: Banco de filtros triangulares espaciados en escalas Mel, Bark y Lineal [7]

- **Obtención de los vectores espectrales**. El siguiente paso consiste en obtener el logaritmo de la envolvente espectral y multiplicar cada componente por 20, para conseguir la envolvente espectral en dB. Tras este paso, se obtiene los vectores espectrales.
- **Obtención de los vectores cepstrales**. Se consiguen aplicando la Transformada de Coseno Discreta (DCT) sobre las componentes espectrales:

$$c_n = \sum_{k=1}^{K} S_k \cdot cos\left[n\left(k - \frac{1}{2}\right)\frac{\pi}{K}\right] \quad n = 1,2,\ldots,L,$$

donde K es el número de coeficientes logarítmicos-espectrales calculados previamente, S_k son los coeficientes logarítmico-espectrales y L es el número de coeficientes cepstrales que se calculan ($L \leq K$). En este punto, se dispone de los coeficientes cepstrales de cada segmento de señal enventanada.

LPC (Codificación Predictiva Lineal)

Es un tipo de codificación que se utiliza en el audio digital y ampliamente utilizado en procesamiento de lenguaje. La idea principal de la **Predicción Lineal** es que una muestra en un determinado instante de la señal de voz se puede aproximar como una combinación lineal de muestras pasadas:

$$\tilde{s}[n] = \sum_{k=1}^{p} \alpha_k \, s[n-k]$$

Donde α_k son los coeficientes LPC, que se obtienen escogiendo los que minimizan la suma del cuadrado de las diferencias entre el valor real s[n] y el valor obtenido por predicción lineal $\tilde{s}[n]$:

$$E = \sum_{n=-\infty}^{\infty} (s[n] - \tilde{s}[n])^2$$

LPCC

Los LPCC (*Linear Prediction Cepstral Coefficients*) derivan de los LPC, que son los coeficientes explicados en el punto anterior. Utilizando los coeficientes α_k, calculados en cada ventana temporal, los LPCC se calculan siguiendo un proceso similar al seguido para calcular los MFCC, tal y como se muestra en la Figura 8.

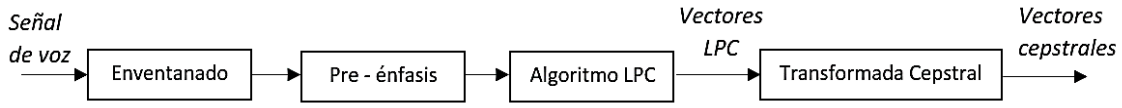


Figura 8: Diagrama de bloques de la parametrización mediante LPCC

LFCC

Los LFCC (*Linear Frequency Cepstral Coefficients*), se obtienen siguiendo el mismo procedimiento que para los MFCC (Ver Figura 6), con la diferencia de que el banco de filtros empleado se encuentra sobre una escala lineal. Los MFCC son el método más ampliamente empleado en los diferentes sistemas de diarización, y los resultados que se obtienen son óptimos en el campo del procesamiento del habla al utilizar la escala de Mel, que da mayor importancia a las frecuencias audibles por el oído humano. Sin embargo, existen estudios [8] que recomiendan usar los LFCCS en algunos casos, ya que algunas características del habla asociadas con el tracto vocal se reflejan mejor en frecuencias más altas.

PLP

PLP (*Perceptual Linear Prediction*) es un método que, junto con MFCC, es ampliamente utilizado en reconocimiento del habla. Los coeficientes PLP son una combinación del **análisis espectral** y del **análisis de predicción lineal** que se calculan siguiendo el procedimiento presentado en la Figura 9. Aunque entre los métodos PLP y MFCC existen muchas similitudes, PLP presenta las siguientes diferencias [9]:

- Se aplica la escala *Bark* en el banco de filtros, a diferencia de la escala Mel utilizada en los MFCC.
- No dispone de una etapa inicial de pre-énfasis. En su lugar, se aplica una etapa *equal-loudness pre-emphasis* (igualación de volumen), que sirve para compensar la diferente percepción de volumen que hay entre diferentes frecuencias. Esta etapa se aplica después del banco de filtros. Después, se aplica una conversión de intensidad – volumen. Tras este procesamiento, se reduce la variación en las amplitudes del espectro.
- Existe una etapa de predicción lineal para obtener los coeficientes cepstrales finales.

El objetivo de PLP es, al igual que en MFCC, dar mayor importancia a la parte del espectro que interesa para procesar la señal de voz.

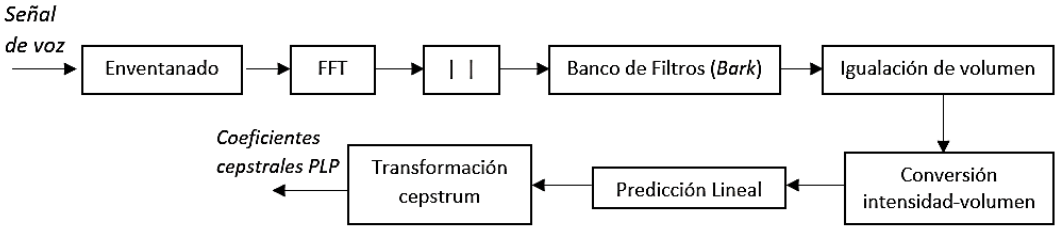


Figura 9: Diagrama de bloques de la parametrización mediante PLP

En la Tabla 1 se resumen los métodos descritos, en función de las técnicas utilizadas para la parametrización y la escala frecuencial para los bancos de filtros:

Método	Técnica	Escala banco de filtros
MFCC	Análisis de espectro	Mel
LFCC	Análisis de espectro	Lineal
PLP	Análisis de espectro + Predicción lineal	Bark
LPC	Predicción lineal	-

Tabla 1: Resumen de las principales características de los métodos de parametrización

3.1.2 Detección de actividad vocal

Consiste en diferenciar los segmentos que contienen voz de los que no contienen voz. Resulta una tarea compleja, debido a que los segmentos de "no voz" pueden deberse a diferentes motivos: silencios, aplausos, ruido de fondo, etc.

La detección de actividad vocal tiene un impacto importante en el funcionamiento del sistema de diarización. Si esta etapa no se realiza, las características acústicas que se extraigan en etapas posteriores serán menos adecuadas debido a que no representan actividad vocal

3.1.3 Segmentación del habla

La segmentación tiene como objetivo detectar los instantes en los que se produce un cambio de locutor. Como resultado de este proceso se obtiene un conjunto de segmentos de audio que contienen voz de un único locutor. Existen varios métodos para realizar esta tarea, que, en general, se clasifican en dos tipos de segmentación:

- **Segmentación basada en métrica (metric-based segmentation)**

  Este tipo de segmentación parte de la base de que cualquier instante del audio de entrada puede ser un punto en el que se produce un cambio de locutor. Para ese instante, se evalúan las ventanas adyacentes (o incluso solapadas entre ellas), para las que pueden ocurrir dos situaciones: que ambas ventanas pertenezcan al mismo locutor y, por tanto, ese instante analizado no corresponda con un punto de cambio o, que, en caso contrario, las dos ventanas pertenezcan a locutores diferentes y en ese caso, el instante analizado sea un punto de cambio. Para poder comparar los dos casos, se han de implementar las dos situaciones (una para el caso de que las dos ventanas corresponden al mismo locutor y otra en la que cada ventana corresponde a un locutor diferente). La solución que más adecuada sea entre las dos determinará si se trata o no de un punto de cambio. El objetivo de este tipo de métodos de segmentación es aplicar un **criterio** o **métrica** que permita identificar cuál de las dos soluciones es la correcta.

  El criterio de decisión más utilizado en los métodos de segmentación basado en métrica es el criterio de Información Bayesiana (*Bayesian Information Criterion, BIC).*

  Bayesian Information Criterion (BIC)

  El criterio BIC es el más ampliamente empleado en la segmentación de locutores. Es una medida que indica la adecuación de un modelo para modelar los datos. En la segmentación, se utiliza como criterio para indicar cuál de las dos soluciones se ajusta mejor: la solución en la que hay cambio de locutor o la solución en la que las dos ventanas corresponden al mismo locutor.

  Debido a que es un criterio que se utiliza en muchos sistemas de diarización, se explica a continuación los pasos a seguir para utilizarlo en la segmentación:

  Sea un segmento X de N muestras, el valor BIC que indica el nivel de ajuste del modelo M al segmento X se define como:

$$\mathrm{BIC}(\mathrm{M}) = \log \mathcal{L}(\mathrm{X}, \mathrm{M}) - \lambda \frac{1}{2} \# (\mathrm{M}) \log(\mathrm{N})$$

  donde $\mathcal{L}(X, M)$ representa la similitud del modelo en los datos, λ es un parámetro dependiente de los datos y # es el número de parámetros que definen el modelo. Por tanto, BIC es un criterio de similitud penalizado por la complejidad del modelo.

  Para utilizar el criterio BIC con el objetivo de identificar si entre dos segmentos hay un cambio, se utiliza la diferencia ΔBIC. Se consideran dos hipótesis: H_0, en la que los segmentos pertenecen un mismo locutor y H_1, en la que los dos segmentos pertenecen a dos locutores diferentes. La expresión para ΔBIC es la siguiente:

$$\Delta BIC = BIC_{H1} - BIC_{H0} = R(i, j) - \lambda P$$

  donde *R(i, j)* es la diferencia de similitudes entre las dos hipótesis y P es un término para introducir penalización por la complejidad del modelo. Por tanto, cuanto más alto sea el valor de ΔBIC implicará que más adecuada es la hipótesis de dos locutores diferentes y cuanto menor sea el valor de ΔBIC, se ajustará mejor la hipótesis de un único locutor.

- **Segmentación basada en modelos (model-based segmentation)**
  Este tipo de métodos se basan en un conocimiento anterior de los datos. Se realiza un conjunto de modelos estadísticos para cada clase acústica (locutor, ruido, música, etc.) que haya en el audio y después, se identifican los segmentos en los que se ha dividido el audio con su modelo correspondiente. Si los locutores en el audio se conocen previamente, se representará cada uno de ellos con un modelo estadístico. Uno de los métodos más populares es el Modelo de Mezclas Gaussianas (*Gaussian Mixture Models*, *GMM*).

En la Tabla 2 se muestra la diferencia principal entre los dos métodos y ejemplos de cada uno de ellos.

Método de segmentación	**Característica**	**Ejemplos**
Basada en métrica	No necesita información previa sobre clases acústicas/locutores en el audio	BIC, Kullback-Leibler, GLR
Basada en modelos	Necesita información previa sobre clases acústicas/locutores en el audio	GMM, Redes neuronales

Tabla 2: Comparativa métodos de segmentación

Muchos sistemas de diarización utilizan un modelo basado en métrica para realizar la segmentación de locutores y después, utilizar los resultados tras la etapa de agrupamiento (que es la etapa que se explica en el siguiente punto) como información previa para hacer una segunda segmentación por medio de un método basado en modelos. Este proceso es conocido como resegmentación y se hace al final del proceso de diarización, para tener mayor exactitud en los límites de cada locutor.

La Figura 10 muestra cómo en un principio no se dispone de información previa sobre tipo de locutores o clases acústicas (la entrada son las características extraídas en la etapa de parametrización). Posteriormente, se han identificado los segmentos que pertenecen al mismo locutor, por lo que se tiene información estadística para cada locutor en el audio. Esa información obtenida sobre los diferentes locutores, es utilizada como información previa para una segmentación basada en modelo en la resegmentación.

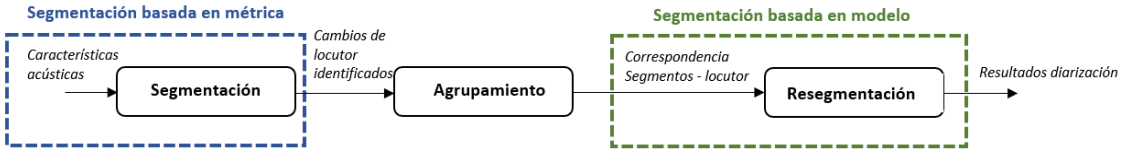


Figura 10: Diarización con etapa de resegmentación

3.1.4 Agrupamiento

El agrupamiento (también conocido como *clustering*), consiste en agrupar todos los segmentos que pertenecen al mismo locutor. Es la etapa siguiente a la segmentación, en la que se han obtenido segmentos que contienen voz de un único locutor. Por tanto, el objetivo del agrupamiento es asociar esos segmentos con los diferentes locutores del audio, agrupando los

segmentos correspondientes a cada locutor en grupos (*clusters*). El número de grupos final corresponde con el número de locutores que existen en el audio.

Existe un gran número de algoritmos de agrupamiento, que además se pueden dividir en varias categorías en función del criterio empleado para la clasificación. En métodos offline, se pueden clasificar en los siguientes: agrupamiento por particiones (*partitioning clustering*) o agrupamiento jerárquico (*hierarchical clustering*).

Agrupamiento por particiones

Requieren que se defina el número de grupos previamente, y agrupar los datos en el número k de grupos que se han definido. Cada grupo tiene al menos un elemento y las muestras de entrada se van agrupando en los diferentes grupos de manera iterativa en función de una métrica o de algún criterio definido. Ejemplos de este tipo de agrupamiento son*: K-Means* y *K-Medoids* [10].

Debido a que es necesario definir el número de grupos, no es idóneo para utilizarlo en un sistema de diarización, ya que normalmente se desconoce el número de locutores que existe en los audios de entrada. Sin embargo, es útil en determinadas aplicaciones, como, por ejemplo, si los audios corresponden a grabaciones de llamadas telefónicas en las que se produce una conversación entre dos personas (dos grupos).

Agrupamiento jerárquico

Es un tipo de agrupamiento en el que los segmentos se van dividiendo o uniendo hasta que se obtiene un número óptimo de grupos. Los dos tipos principales de este tipo de agrupamiento son el agrupamiento jerárquico **aglomerativo** (*agglomerative hierarchical clustering*) y el agrupamiento jerárquico **divisivo** (*divisive hierarchical clustering)*.

- **Agrupamiento aglomerativo:** En la situación inicial, hay un número elevado de grupos y se van agrupando hasta obtener el número de grupos óptimo.
- **Agrupamiento divisivo:** En la situación inicial, existe un único grupo y se va dividiendo hasta alcanzar el número de grupos óptimo.

En la implementación de ambos agrupamientos, se debe definir dos elementos: la distancia y el criterio de parada. La **distancia** entre grupos indica la diferencia entre grupos y el **criterio de parada** determina el momento de parar el proceso iterativo que une/separa las agrupaciones por haber alcanzado el número óptimo de grupos.

Una correcta elección de la distancia y el criterio de parada es importante debido a que puede dar lugar a que el número de grupos identificado sea menor o mayor que el número real de locutores existentes.

El agrupamiento es un proceso iterativo en el que se van uniendo/separando los grupos tomando como criterio la distancia entre los grupos. En el caso del agrupamiento aglomerativo, que es el más empleado, los grupos se van uniendo en cada etapa. En la etapa k, se unen los dos grupos que tengan menor distancia entre ellos, que serán por tanto los dos grupos más similares, creando un nuevo grupo que tendrá nuevos valores de distancia respecto al resto de grupos. En la etapa k+1, se comprueba el criterio de parada, analizando si el mínimo valor de distancia es superior que el criterio definido. Si se cumple la condición, se termina el proceso y el número de

grupos independientes que hay en esa etapa es el número de grupos finales. En caso de no cumplirse la condición, se repite el proceso.

Entre las distancias que se emplean como medida de diferencia entre los grupos, las más empleadas son las que se usan en la fase de segmentación. Entre ellas BIC es ampliamente utilizada como medida. Se calcula la diferencia ΔBIC entre todos los pares de grupos existentes y se une el par cuyo valor de ΔBIC haya sido menor.

Número óptimo de grupos

Determinar el número óptimo de grupos es uno de los puntos principales del proceso de diarización, ya que indica el número de locutores diferentes que existen. En el agrupamiento jerárquico aglomerativo, se obtiene cuando el proceso iterativo cumple el criterio de parada y el número de grupos en esa etapa es el óptimo.

Otra alternativa empleada por algunos sistemas de diarización es determinar previamente un número N de grupos, mayor que el número de locutores (no se sabe el número de locutores existentes, pero en función de la duración del audio se puede tener una idea aproximada y determinar un valor más alto). El procedimiento consiste en realizar el proceso iterativo completo, sin criterio de parada, las N veces obteniendo N soluciones diferentes en las que cada solución tiene un número de grupos diferente. Posteriormente, se identifica la solución óptima entre todas las soluciones en base a algún criterio establecido.

Los dos tipos de agrupamientos principales, junto con sus características más importantes y ejemplos de cada uno de ellos se resumen en la Tabla 3.

Agrupamiento	Número de grupos	Funcionamiento	Ejemplos
Particiones	Definido previamente	Los datos se van asociando en función de una medida a los grupos definidos	K-Means, K-Medoids
Jerárquico	No está definido define previamente	Los datos se van uniendo/separando según una medida que determine la similitud entre grupos	*Aglomerativo*: Los grupos se van uniendo *Divisivo*: Los grupos se van separando

Tabla 3: Comparativa tipos de agrupamientos

3.1.5 Resegmentación

La etapa de resegmentación es una etapa que no se encuentra en todos los sistemas de diarización. Como se ha explicado en el apartado de segmentación, se suele emplear en aquellos sistemas que implementan un modelo de segmentación basado en métrica debido a que no se conocen previamente las clases acústicas o las características de los locutores en el audio de entrada. Por ello, emplea la información sobre los locutores que el sistema ha obtenido durante el proceso para hacer modelos estadísticos de cada locutor y emplear esa información para realizar una segmentación basada en modelo. El objetivo de esta etapa final es ajustar los límites en los que se produce un cambio de locutor.

3.2 Resultados de la diarización

Una vez realizado el proceso de diarización, se han obtenido el número de locutores existentes en el audio y los instantes en los que habla cada locutor. Para guardar los resultados, la manera más habitual es utilizar el formato RTTM (*Rich Transcription Time Marked*). Este formato fue propuesto por el NIST en sus evaluaciones [11] [12] [13], y es el más utilizado por los diferentes sistemas de diarización.

Los ficheros con formato RTTM son ficheros de texto en los que cada línea corresponde a un turno en el que habla cada diferente locutor. Los diferentes campos que contiene cada línea del fichero RTTM, están separados por espacios y son los que se muestran en la Tabla 4.

1	2	3	4	5	6	7	8	9	10
Tipo	ID fichero	ID Canal	Inicio Turno	Duración Turno	Campo ortográfico	Tipo locutor	Nombre Locutor	Nivel confianza	Tiempo de "Lookahead"

Tabla 4: Nombre de los campos del RTTM

Tipo: Tipo del objeto. Por lo general, en tareas de diarización tiene el valor "SPEAKER".

ID fichero: Nombre del fichero (sin extensión).

ID Canal: Canal del audio, valor "1" o "2".

Inicio Turno: Instante en el que comienza el turno del locutor, medido en segundos desde el inicio del fichero.

Duración Turno: Duración del turno en segundos.

Campo ortográfico: Representación ortográfica, utilizado para tarea de conversión habla a texto.

Tipo locutor: Tipo de locutor del turno.

Nombre Locutor: Nombre del locutor del turno.

Nivel de confianza: Indica la probabilidad de que la información sea correcta.

Tiempo de "Lookahead" de la señal (Signal Look Ahead Time): Instante de la última muestra de la señal del audio, utilizada para determinar los valores en los campos del objeto RTTM.

Es común que muchos de los ficheros RTTM que contengan información del proceso de diarización tengan campos con el valor "NA", para indicar que ese campo no es relevante o no se dispone de información. En la Figura 11, se muestra parte de un fichero RTTM obtenido tras la diarización en un archivo de audio de la base de datos, donde la información más relevante es el instante en el que comienza el turno de cada locutor, su duración y el locutor asignado a ese intervalo.

```
SPEAKER 20091121_13532200_0002531371_001_001_BARATZE_EKOL_1-Non-Silenced-Audio 1 0.0 13.37 <NA> <NA> speaker1 <NA>
SPEAKER 20091121_13532200_0002531371_001_001_BARATZE_EKOL_1-Non-Silenced-Audio 1 13.37 13.62 <NA> <NA> speaker2 <NA>
SPEAKER 20091121_13532200_0002531371_001_001_BARATZE_EKOL_1-Non-Silenced-Audio 1 26.99 7.13 <NA> <NA> speaker1 <NA>
SPEAKER 20091121_13532200_0002531371_001_001_BARATZE_EKOL_1-Non-Silenced-Audio 1 34.12 4.98 <NA> <NA> speaker3 <NA>
```

Figura 11:Resultado de diarización en formato RTTM

3.3 Evaluación de la diarización de locutores

En este apartado se describe el criterio para la evaluación de los resultados obtenidos tras la diarización de locutores. A pesar de que hay medidas que determinan el error cometido en las etapas de segmentación y agrupamiento de manera independiente, el criterio utilizado para evaluar los resultados en la diarización de locutores es el Error de Diarización (***Diarization Error Rate, DER***). El DER es una medida definida por el NIST para poder comparar los resultados obtenidos con la información de referencia e identificar cuánto porcentaje del tiempo el sistema no ha identificado correctamente al locutor. El DER es la suma de tres errores:

$$DER = E_{SPK} + E_{FA} + E_{MISS}$$

- ***E_{SPK} (Speaker Error):*** Porcentaje del tiempo en el que se ha identificado a un locutor incorrecto.
- ***E_{FA} (False Alarm):*** Porcentaje del tiempo en el que se detecta voz cuando en la referencia no hay voz.
- ***E_{MISS} (Missed Speaker):*** Porcentaje del tiempo en el que no se detecta voz cuando en la referencia hay voz.

Evaluándolo en función del tiempo en el que se produce cada error, el DER se define como:

$$DER = \frac{SPK + FA + MISS}{TOTAL}$$

Donde *SPK*, *FA* y *MISS* es la duración acumulada del tiempo que se identifican locutores erróneos, falsas alarmas y locutor no identificado, respecto a la duración total.

En la evaluación de los resultados, se suele definir un intervalo alrededor del instante en el que se produce el cambio de locutor, en el que no se tiene en cuenta el error cometido. Este valor, conocido como ***collar*** en inglés, permite que la evaluación de los resultados no se vea afectada en exceso por posibles inexactitudes en el marcado de los instantes en los que se produce el cambio y comúnmente tiene un valor de $\pm$ 250ms.

4. Objetivos y alcance del trabajo

Este trabajo tiene como objetivo procesar la base de datos multilocutor de EITB para que sea adecuada en el desarrollo un sistema TTS. Se estudiarán los sistemas de diarización existentes para poder identificar cuál de ellos es el más adecuado para la base de datos y se implementarán para obtener los audios modificados de manera que puedan ser utilizados en un sistema TTS. Los objetivos que se han de cumplir son los siguientes:

- *Análisis de la base de datos:* Distinguir entre toda la información existente en la base de datos, la información útil para poder desarrollar un sistema TTS.
- *Estudio e implementación de sistemas de diarización:* Es la parte más importante del trabajo. Consiste en seleccionar de entre los sistemas de diarización existentes, cuáles se ajustan mejor a las características de la base de datos utilizada. Además, se debe encontrar la mejor configuración de los sistemas de diarización seleccionados.
- *Evaluación:* Tras la configuración de los sistemas de diarización, es necesario una parte de evaluación, que permita conocer la precisión de los resultados obtenidos y el rendimiento de cada sistema con la base de datos.
- *Automatización:* Tras la optimización de los sistemas de diarización seleccionados, el proceso se debe realizar de manera automatizada para la cantidad de ficheros que se indique. Así, se podrá utilizar en la base de datos del trabajo y en otras bases de datos que contengan ficheros de audio.

Al finalizar el trabajo, se habrá comprobado el rendimiento de los sistemas de diarización seleccionados en ficheros de audio de la base de datos. Esto permitirá conocer la adecuación de estos sistemas para poder aplicarlos de manera general en grabaciones en las que existen varios locutores, lo que permitirá extraer la voz del locutor de interés y utilizarlas en el desarrollo de sistemas TTS.

5. Beneficios

En este apartado se detallan los beneficios del proyecto realizado. Los beneficios se dividen en tres apartados: técnicos, sociales y económicos.

5.1 Beneficios técnicos

Los beneficios técnicos son los más importantes del trabajo realizado. La adecuación de una base de datos es una parte fundamental para poder desarrollar un sistema TTS. Disponer de un conjunto de ficheros de audio con únicamente las partes de voz correspondientes al periodista junto con su transcripción en texto, cumple la primera fase en el desarrollo de un sistema TTS.

Este hecho aporta una serie de ventajas: por un lado, no es necesario que nuevos locutores graben fragmentos de conversaciones con sus voces, lo que supone una reducción importante en el tiempo total requerido para el desarrollo del sistema. Por otro lado, una segunda ventaja viene dada por las características de la base de datos original, debido a que consiste en grabaciones realizadas por periodistas de EITB. Por tanto, las condiciones necesarias de calidad, entonación e inteligibilidad de la voz están garantizadas.

Además, la solución final permitirá que la diarización de locutores se pueda implementar en otras bases de datos que contengan voz mezclada de varios locutores (por ejemplo, grabaciones del parlamento, vídeos del Gobierno Vasco, etc.) para obtener más contenido en condiciones de ser usado en sistemas TTS.

5.2 Beneficios sociales

El principal beneficio social de este trabajo es que permite que un futuro se pueda desarrollar un sistema TTS de calidad tanto en castellano como en euskera. Disponer de un TTS en el idioma nativo aporta una serie de beneficios sociales que puede mejorar la calidad de vida de muchas personas, como puede ser la comunicación oral para gente sin voz o con patologías orales o la lectura de libros y páginas web para personas sin vista.

5.3 Beneficios económicos

El aspecto económico no es el principal aspecto a considerar durante el desarrollo del proyecto. Sin embargo, tras adecuar la base de datos empleada en este trabajo, se dispone de ella para cualquier tarea en el futuro, sea desarrollar un sistema TTS o para cualquier otro objetivo. El hecho de no tener que hacer la grabación de las voces de los locutores previamente, supone un ahorro en medios técnicos para la adquisición de los ficheros de audio.

Adicionalmente, tener el proceso automatizado permite modificar parámetros en los sistemas utilizados de manera simultánea en el conjunto de ficheros, lo que supone un ahorro tanto en tiempo como en recursos empleados.

6. Descripción de requerimientos

Los requerimientos se pueden dividir en dos aspectos principales: los requerimientos de la base de datos sobre la que se va a realizar la diarización de locutores y los de los sistemas de diarización.

6.1 Base de datos

Debe reunir las siguientes características:

- *Cantidad de información suficiente*. Cuanto mayor sea el número de ficheros de audio disponibles, mejor será la solución final. Por un lado, mayor será el número de oraciones y términos empleados para desarrollar el sistema TTS. Por otro lado, los sistemas de diarización tendrán mayor variedad de audios para comprobar su funcionamiento y evaluar sus resultados.
- *Variedad en los ficheros*. Los ficheros de audio deben tener diferentes características en aspectos como el número, género y entonación de los locutores. De esta manera, se puede comprobar el rendimiento de los sistemas de diarización en diferentes situaciones.
- *Calidad de los ficheros*. Las voces grabadas en los ficheros deben ser inteligibles y tener una intensidad y entonación adecuadas.
- *Transcripciones en texto*. Es necesario disponer de las transcripciones en texto de los fragmentos de audio del periodista, para que el sistema TTS futuro pueda asociar las palabras en audio con las palabras en texto.

6.2 Sistemas de diarización

Los sistemas de diarización empleados en los ficheros de audio de entrada deben cumplir las siguientes condiciones:

- *Sistema offline*. El proceso de diarización debe realizarse de manera *offline*, es decir, actuando sobre ficheros grabados previamente. Los sistemas *offline* ofrecen mejores resultados que los sistemas *online*, que realizan la diarización en tiempo real, y, por tanto, están expuestos a mayor error.
- *Múltiples locutores*. Se deben poder aplicar a ficheros de audio en los que hay más de dos locutores, sin que ello provoque un aumento importante en el error cometido.
- *Rapidez*. En la medida de lo posible, los sistemas deben obtener los resultados de diarización en el menor tiempo posible, utilizando en cada etapa las técnicas que menor tiempo de procesamiento requieran.
- *Neutralidad con el idioma empleado*. Los resultados no deben estar condicionados al idioma utilizado en los ficheros de audio.

7. Análisis de alternativas

En este apartado se exponen los criterios seleccionados en la elección del sistema de diarización y la importancia de cada criterio en la elección final. Por otro lado, se describen las alternativas estudiadas y su utilidad para emplearlas en la base de datos.

7.1 Criterios de selección

El principal punto donde se ha debido escoger entre diferentes alternativas es en la elección de los sistemas de diarización. Se han tenido en cuenta los siguientes aspectos en la elección:

- ***Rendimiento:*** La mejor manera de comparar las prestaciones de cada sistema es por su error cometido en la diarización (DER). Por ello, un sistema será más adecuado para los audios de entrada cuento menor sea el DER cometido. Además, dentro de este apartado hay algunos aspectos importantes que se tienen que valorar:
    - ***Parametrización:*** Consiste en extraer las características que representan la señal de voz. Los diferentes sistemas pueden emplear técnicas diferentes o la misma técnica con parámetros distintos.
    - ***Número de grupos:*** Es importante que el sistema pueda identificar todos los posibles locutores que pueda haber en las grabaciones, haciéndolo de manera automática y sin tener que definirlo previamente.
    - ***Algoritmo de agrupamiento:*** Es una de las partes más importantes del proceso. Existen varios algoritmos que se pueden emplear para determinar qué partes del audio pertenecen a un mismo grupo/locutor, y elegir el algoritmo adecuado permitirá que el número de locutores presentes en el audio se identifique correctamente.
- ***Sencillez de implementación:*** se debe tener en cuenta la facilidad de implementación de los diferentes sistemas. Un sistema más sencillo de implementar, que ofrezca resultados similares a otro que necesite más tiempo en su implementación, es mejor alternativa.
- ***Documentación:*** disponer del funcionamiento del sistema documentado (con un manual, guía o vídeo explicativo), los requerimientos para su implementación y el formato de entrada y salida de los datos, es un aspecto a tener en cuenta en el estudio de los diferentes sistemas de diarización. En caso de no disponer de esa información, aumenta considerablemente el tiempo dedicado a implementar cada sistema.
- ***Lenguaje de programación:*** sin ser un criterio decisivo en la elección del sistema, se debe considerar, ya que puede facilitar la parte de optimización del sistema. Por lo general, la mayoría de sistemas están desarrollados en Python, aunque existen algunas alternativas en MATLAB o Java.

7.2 Alternativas

Actualmente, existe una gran variedad de alternativas para llevar a cabo el proceso de diarización de locutores. La mejor manera de disponer de un gran número de implementaciones junto con información sobre su uso y requerimientos, es en el repositorio de *GitHub*[4]que reúne

[4] https://github.com/wq2012/awesome-diarization#Software

varios sistemas de diarización. A continuación, se describen las propuestas más adecuadas para utilizar en la base de datos del trabajo:

SIDEKIT for diarization(s4d)

S4d es una herramienta que permite realizar el proceso de diarización de manera sencilla y con buenos resultados. El usuario puede utilizarlo en todas las fases de proceso, desde el procesamiento del audio inicial hasta el análisis del rendimiento del sistema.

pyAudioAnalysis

Se trata de una librería de Python que puede ser aplicada en múltiples tareas de análisis de audio. Permite extraer características de audio como coeficientes MFCC y espectrogramas, clasificar audios, detectar silencios, segmentar audios y también diarización de locutores. Ofrece la ventaja de que sus aplicaciones son independientes entre ellas, por lo que, si la diarización no cumple con los objetivos esperados, se puede emplear en la parte de procesado de los audios iniciales.

pyBK

La particularidad de este método es que emplea un modelo de "llaves binarias" o *binary keys* que permite representar las características acústicas de los audios de entrada de manera más eficiente que otras alternativas. Se dispone de la implementación completa para el proceso de diarización: segmentación, agrupamiento y resegmentación. Además, es posible configurar parámetros en cada etapa del proceso, lo que permite optimizar el sistema en las diferentes fases para adecuarlo a la base de datos.

Kaldi

Es un conjunto de herramientas de reconocimiento de voz escrito en C++. Pone a disposición del usuario toda la documentación necesaria, con ejemplos y librerías, aunque está más orientado al reconocimiento del habla y de locutores que a la diarización.

Pyannote.audio

Es una herramienta escrita en Python para la diarización de locutores y que utiliza la librería de aprendizaje automático PyTorch. Dispone de modelos entrenados para aplicaciones como la detección de voz, cambio de locutor y detección de solapamiento de locutores. Es la alternativa que más tiempo requiere.

Existen varias alternativas adicionales a los métodos descritos anteriormente, pero tras el estudio de la documentación sobre el funcionamiento e implementación de cada una de ellas, se han descartado fundamentalmente por las siguientes razones:

- *Dificultad de implementación*: Sistemas que tienen una gran complejidad y que no son los más adecuados para el caso de este trabajo. Adicionalmente, se descartan métodos *online* que realizan el proceso de diarización en tiempo real ya que su implementación es más compleja y producen peores resultados. Al disponer de una base de datos con las grabaciones guardadas en ficheros de audio, se requiere que la diarización sea *offline*.
- *Sistemas diseñados para datasets específicos*: Algunos de los métodos se han diseñado para poder emplearlos con un *dataset* o base de datos concreta. Por ello, si bien se puede

modificar el sistema para adaptarlo a los ficheros utilizados en este trabajo, no se garantiza que los resultados vayan a ser igual de óptimos que con el *dataset* original.

A continuación, en la Tabla 5, se muestra la evaluación de cada sistema en los criterios de selección descritos en el apartado anterior:

	S4d	**pyAudioAnalysis**	**Kaldi**	**pyannote.audio**	**pyBK**
Rendimiento (60%)	9	6	6	9	9
Sencillez de implementación (20%)	7	9	7	5	7
Documentación (15%)	9	9	9	9	9
Lenguaje programación (5%)	9	9	7	9	9
Total	**8.6**	7.2	6.7	8.2	**8.6**

Tabla 5: Criterios de selección de sistema de diarización

La evaluación de los diferentes sistemas con los criterios de selección establecidos, indican que **s4d** (SIDEKIT for diarization) y **pyBK** (binary key) son los sistemas que mejores prestaciones ofrecen para la realizar la diarización. Por ello, se ha realizado el estudio, implementación y adecuación de ambos sistemas para la diarización de la base de datos de EITB. La selección de dos métodos, permite comparar el rendimiento de ambos y determinar cuál es el más adecuado para utilizar en la base de datos.

8. Descripción de la solución

Una vez estudiados y seleccionados los métodos a emplear en la parte de diarización, se describe la metodología seguida para obtener la solución completa, cumpliendo con los objetivos descritos previamente.

En primer lugar, se describen las características de la base de datos utilizada y los ajustes realizados para adecuarla a los sistemas de diarización que se han empleado.

Posteriormente, se detallan los dos sistemas de diarización de locutores empleados en este proyecto, describiendo su funcionamiento y los resultados obtenidos con cada uno de ellos en los audios de entrada.

Por último, se evalúan los resultados obtenidos con los dos sistemas de diarización para analizar si son adecuados para la diarización en los ficheros de la base de datos y determinar si alguno de los dos métodos ofrece mejor rendimiento.

8.1 Base de datos

La base de datos empleada consiste en ficheros de audio correspondientes a noticias de EITB que contienen la voz del periodista junto con cortes de personas entrevistadas o relacionadas con la noticia que se describe. Debido a que las voces grabadas corresponden a periodistas profesionales, cumplen las condiciones de inteligibilidad y calidad necesarias. Por otro lado, además de castellano, las grabaciones están en euskera, que es un idioma para el que la cantidad de material disponible para desarrollar sistemas TTS es mucho menor.

En la base de datos se puede distinguir el siguiente contenido, que se ha clasificado en función de su validez para emplearlo en el desarrollo de un sistema TTS:

- **Ficheros con voz del periodista y voz de personas entrevistadas**. Son los ficheros de audio que contienen la parte del periodista junto con intervenciones de las personas entrevistadas. Estos ficheros son los que se van a utilizar en el sistema de diarización, con el objetivo de separar los diferentes locutores y extraer la parte correspondiente al periodista. La base de datos contiene las transcripciones de muchos de los ficheros de audio, con la parte hablada por el periodista en texto. Por ello, disponer de audios que contengan únicamente la voz del periodista junto con su transcripción en texto, permite que se puedan utilizar para desarrollar un sistema TTS.
- **Ficheros con únicamente voz de personas entrevistadas.** Estos ficheros son en general cortos, con partes que únicamente corresponden a personas entrevistadas. No se emplean en la diarización, debido a que no contienen voz del periodista.
- **Ficheros con únicamente voz del periodista.** Estos ficheros no se emplean en la diarización, ya que no contienen voces de diferentes locutores. Sin embargo, son útiles para desarrollar el sistema TTS al corresponder a la voz del periodista. Por tanto, se realiza el procesado inicial para extraer los silencios y el fichero resultante será válido para utilizarlo en el TTS.

Eliminación de silencios

En el proceso de diarización, es necesario eliminar silencios de los ficheros de audio, tanto los del inicio y final como los silencios intermedios. Esta tarea puede ser incluida dentro del proceso de diarización o realizarse previamente. Se ha optado por hacer la eliminación de silencios antes de la diarización, debido a que de esta manera se hace una única vez y no se requiere repetir el proceso en cada sistema de diarización utilizado.

La eliminación de silencios en los ficheros se lleva a cabo en una etapa conocida como *VAD (Voice Activity Detection).* VAD determina en la señal de audio de entrada qué instantes contienen voz y cuáles silencios. Su implementación se realiza determinando si hay "voz" o" no voz" como una decisión binaria y utilizando un umbral para determinar el nivel de precisión del proceso.

py-webrtcvad

Entre las diferentes alternativas existentes para realizar la eliminación de silencios, se ha optado por una solución [14] que hace uso de ***py-webrtcvad***[5], debido a su sencillez de implementación y rapidez. *Py-webrtcvad* permite establecer el nivel de "agresividad", que es un parámetro que define lo estricto que va a ser el sistema eliminando las partes del audio de entrada que no contienen voz. Se define con un valor entero entre 0 y 3, siendo el 0 el menos agresivo y 3 el más agresivo. El valor escogido ha sido 3, debido a que tanto en los sistemas de diarización como en el desarrollo de un sistema TTS, únicamente es necesaria las partes de audio que contienen voz. En las siguientes imágenes se puede observar el audio original (Figura 12) y el audio resultante tras eliminar silencios con un nivel de "agresividad" 3 (Figura 13).

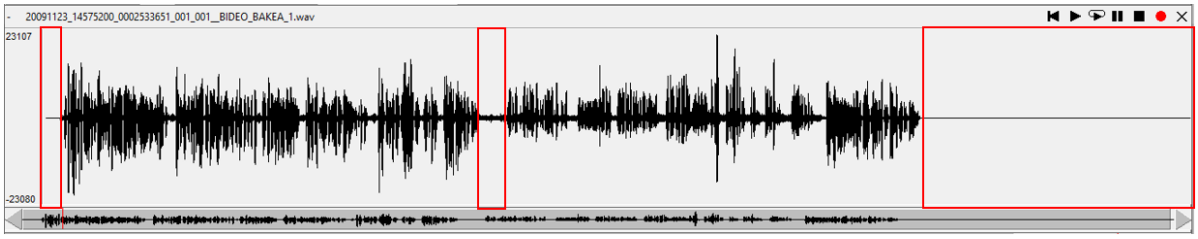


Figura 12: Audio original

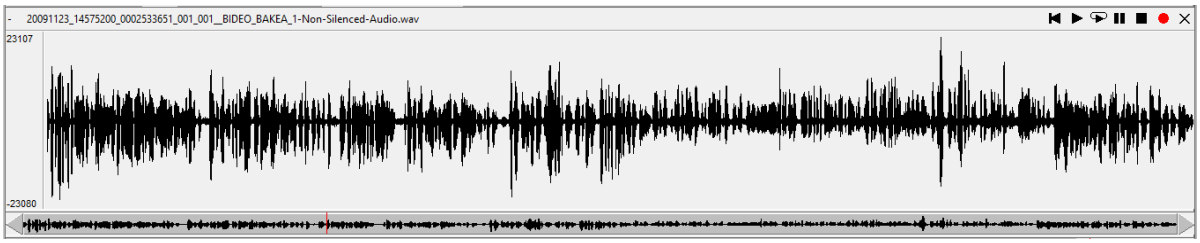


Figura 13: Audio con silencios eliminados

[5] *https://github.com/wiseman/py-webrtcvad*

8.2 Estudio de sistemas de diarización de locutores

En este apartado se detallan los métodos que se han elegido para realizar la diarización en los ficheros de la base de datos. Los dos métodos utilizados en este proyecto para la diarización de locutores son el método que emplea ***binary keys*** (llaves binarias) y el método que utiliza SIDEKIT for Diarization o ***s4d***.

Ambos métodos han sido implementados y evaluados previamente con ficheros de audio de diferentes características en las evaluaciones [21] y [24], por lo que son métodos adecuados para utilizarlos este trabajo. El objetivo es aplicar estos métodos en los ficheros de audio de la base de datos de EITB, aplicando la configuración original y diferentes modificaciones para determinar la configuración óptima.

Los dos métodos ofrecen el código utilizado en sus evaluaciones para que el usuario pueda replicarlas. La diferencia en la implementación de ambos sistemas es la siguiente: en el caso de la implementación del modelo que utiliza *binary keys*, se ha obtenido desde su repositorio de github[6], en donde están los scripts y ficheros necesarios para realizar la implementación. Por otro lado, para la implementación de s4d, hace uso de SIDEKIT, que es un conjunto de herramientas de acceso libre para reconocimiento de locutores y que para el caso concreto de diarización de locutores, dispone de *s4d* (*SIDEKIT for Diarization*). En ambos casos la implementación estaba realizada en Python.

A continuación, se describen los dos métodos empleados, por medio del siguiente orden: en primer lugar, se describe el método y sus características principales. Posteriormente, se describe la configuración del estudio original y los resultados obtenidos en la base de datos de EITB con esa configuración. Finalmente, se prueban diferentes configuraciones para encontrar la más adecuada, en función del error obtenido en la diarización.

8.2.1 Diarización de locutores con binary keys

Este método tiene como característica principal que representa los segmentos por medio de vectores binarios. Estos vectores contienen las características específicas de cada locutor y permiten su comparación calculando medidas de similitud entre ellos. La principal ventaja de este método es su bajo coste computacional.

Debido a que el método ha tenido una serie de mejoras sobre su diseño original, se describe el método de la siguiente manera. Inicialmente, se describen las características principales del método, sobre su diseño original. Después, se muestran las mejoras iniciales sobre ese diseño, para concluir con las mejoras más recientes que se han implementado en algunas etapas de su proceso de diarización.

Este método entrena un modelo KBM (*Binary Key Background Model*) que luego utiliza para convertir las características acústicas en un vector de valores binarios. Como paso previo a explicar el modelo KBM, se describe brevemente el modelo GMM-UBM (*Gaussian Mixture Model – Universal Background Model*).

[6] https://github.com/josepatino/pyBK

GMM – UBM

Los modelos probabilísticos GMM son comúnmente utilizados en reconocimiento y diarización de locutores debido a su capacidad de adaptarse a la variabilidad de la señal de voz. Los GMM se definen por un número de K componentes gaussianas multiplicadas por el peso que tienen w_k. Un modelo GMM λ se expresa por su función de densidad de probabilidad para una observación x de la siguiente forma:

$$p(x|\lambda) = \sum_{k=1}^{K} w_k \mathrm{N}(x|\mu_k, \Sigma_k)$$

donde w_k es la probabilidad asociada a cada componente gaussiana de forma que $\sum_{k=1}^{K} w_k = 1$ y μ_k y Σ_k son el vector de media y la matriz de covarianza de cada componente gaussiana k.

Por tanto, para un conjunto de N vectores de características acústicas $\mathrm{X} = X_{1,\ldots,} X_N$ en un segmento de voz, su log – similitud a un modelo GMM λ, se define como:

$$\log\big(p(\mathrm{X}|\lambda)\big) = \sum_{n=1}^{N} \log\big(p(x_n|\lambda)\big)$$

La adaptación de los GMMs a las observaciones acústicas se realiza por medio del algoritmo de Maximización de la Esperanza (*Expectation Maximization, EM*), que ofrece buenos resultados cuando se dispone de gran cantidad de datos [15]. Sin embargo, ese no siempre es el caso cuando se trabaja con segmentos de audio en reconocimiento y diarización de locutores. Para dar solución a ese problema, se emplea un modelo UBM [16]. El objetivo es crear un modelo estadístico que represente las características acústicas de manera genérica e independiente del locutor, para después comparar las características acústicas específicas de los locutores sobre ese modelo y poder determinar las similitudes entre ellos.

Sistema de diarización mediante binary keys

La particularidad de este método es el empleo de *binary keys*, una técnica innovadora que representa las frases pronunciadas por los locutores por medio de vectores binarios. La principal ventaja de esta técnica es su menor coste computacional en la comparación de locutores.

El sistema se puede dividir en dos módulos principales, como se puede ver en su arquitectura en la Figura 14. El sistema consiste en un primer módulo que convierte el espacio acústico de entrada en una representación binaria que contenga las características de los locutores (Módulo de Procesamiento Acústico) y, por otro lado, un módulo para realizar un agrupamiento AHC con la información binaria del módulo anterior (Módulo de Procesamiento Binario).

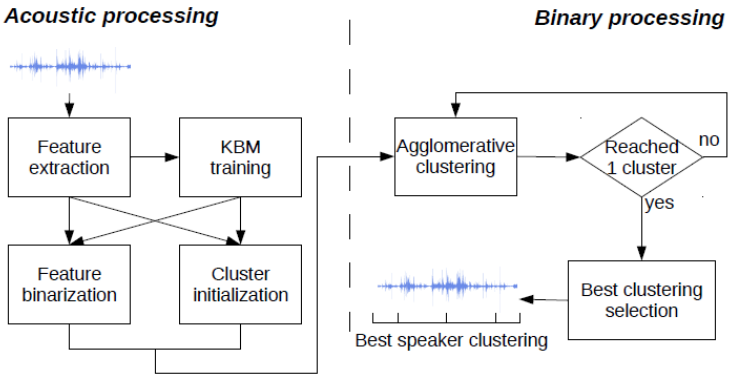


Figura 14: Esquema del sistema de diarización con técnica de binary keys [4]

Módulo de Procesamiento Acústico

Este bloque convierte las características acústicas de los vectores de entrada en los *binary keys*. El resultado de esta transformación es un modelo acústico de tipo UBM denominado KBM (*Key Background Model).* El KBM se obtiene tras entrenar los datos para representar las características acústicas con GMM (Modelos de Mezclas Gaussianas).

- ***Entrenamiento de KBM para diarización de locutores***

En la diarización de locutores, no se dispone de información previa sobre las identidades de los locutores, por lo que el KBM es resultado de un método que no requiere de información externa, introducido en [17].

La etapa inicial del algoritmo de entrenamiento del KBM consiste en extraer un conjunto de gaussianas únicas de las características de entrada, seguido de un proceso iterativo para seleccionar las componentes más complementarias y discriminativas, con el objetivo de representar el espacio acústico completo del habla. Una vez se haya obtenido el conjunto de gaussianas, se seleccionan las componentes de manera iterativa hasta alcanzar el número de gaussianas que se ha definido.

En primer lugar, la obtención de gaussianas inicial se realiza utilizando una ventana de tamaño fijo, junto con un desplazamiento y solapamiento. El valor del desplazamiento se define en función del tamaño de la señal de entrada, con el objetivo de obtener cientos de gaussianas (desplazamiento mayor cuanto mayor sea la duración de la señal de entrada).

Una vez obtenido el conjunto de gaussianas, el siguiente paso es seleccionar las gaussianas más discriminativas de forma iterativa, hasta tener N gaussianas, siendo N un valor definido previamente. Como paso previo al proceso iterativo, se debe seleccionar la primera gaussiana, que será la que mejor modele el segmento del que se ha extraído:

$$\arg\max_{i} Lkld(s_{i.}, \theta_i)$$

donde θ_i es la gaussiana entrenada para el segmento i y *Lkld*(*likelihood*) determina con qué probabilidad se ajusta la gaussiana i al segmento i. Para la selección iterativa de las gaussianas, se establece un vector v_{KL2} disimilitud global, que represente las distancias entre las gaussianas ya seleccionas en cada etapa respecto a todas las gaussianas restantes. Este vector se inicializa

a ∞ debido a que al principio ninguna componente ha sido seleccionada. Tras la selección de la primera gaussiana e inicialización del vector, el proceso iterativo es el siguiente:

1. Se calcula la divergencia KL2 (Kullback–Leibler Simétrica) entre la gaussiana seleccionada en la iteración anterior θ' y el resto de gaussianas que no se han seleccionado θ_k y se guarda el valor de disimilitud en el vector v_{KL2} en la posición correspondiente a cada gaussiana:

$$v_{KL2}[j] = \min\left(v_{KL2}[j], KL2(\theta', \theta_j)\right)$$

2. Se añade al KBM la gaussiana θ^k que tenga la mayor diferencia con aquellas gaussianas ya seleccionadas previamente:

$$arg\max_k(v_{KL2}[k])$$

   De esta forma, se va añadiendo en cada iteración al KBM la gaussiana más discriminativa.
3. Vuelve al paso (1) hasta que se haya obtenido el número N de componentes del KBM.

Este procedimiento para la selección de las gaussianas que forman el KBM se puede ver en la Figura 15.

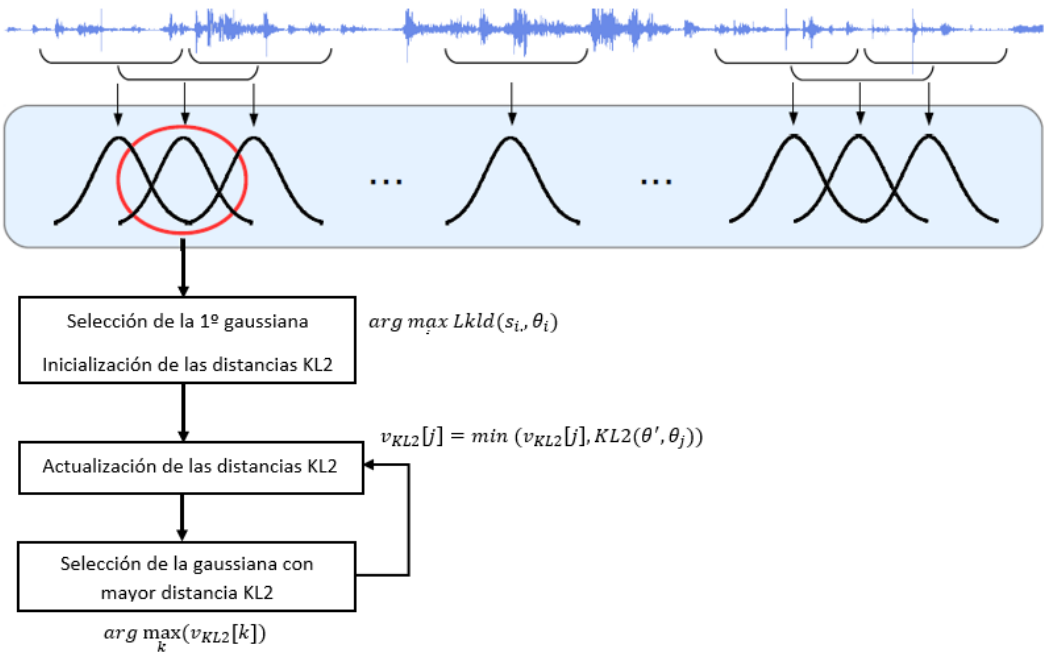


Figura 15: Proceso iterativo para la selección de las gaussianas. Imagen adaptada de [4]

La principal ventaja respecto al modelo GMM, es que los BKs resultantes utilizando KBM son mucho más discriminativos. Esto se debe a que GMM modela el espacio acústico completo, mientras que las componentes del KBM priorizan el habla. Además, tienen un menor coste computacional que el algoritmo EM utilizado con GMM.

- ***Conversión de datos de entrada en binary keys***

En esta etapa, la señal de entrada se divide en segmentos de igual duración (con solapamiento entre ellos) y cada segmento se convierte en un *binary key (BK)*. El proceso de conversión para obtener los BK se puede ver en Figura 16.

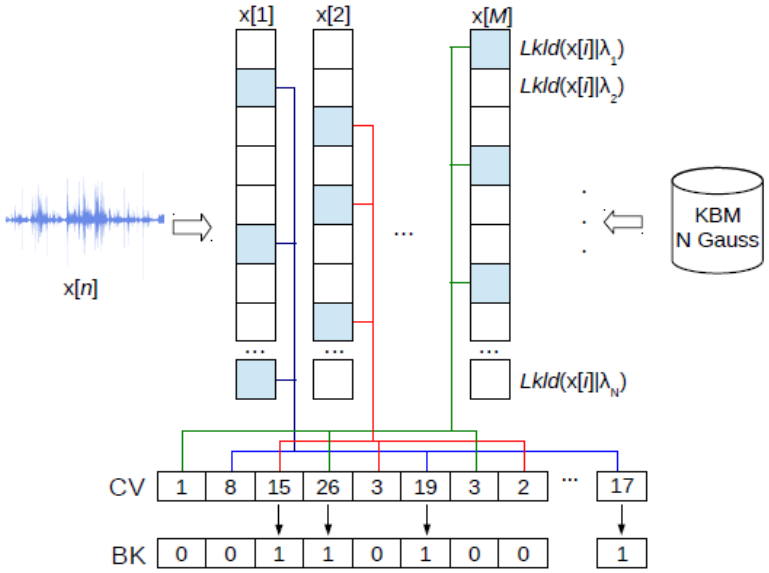


Figura 16: Ejemplo del proceso de extracción de los Binary Keys [19]

Un BK v_f = {v_f [1], …, v_f [N]} con v_f *[i]* = {0,1} es un vector binario de dimensión N, siendo N el número de componentes en el KBM obtenido en la etapa anterior. La posición i del v_f representa la gaussiana i del KBM. Si el valor de v_f en la posición i es verdadero, $v_f[i]$ = 1, indica que la gaussiana en la posición i coexiste en un mismo área del espacio acústico que los datos que se modelan. Es decir, si $v_f[i]$ = 1, la gaussiana i del KBM tiene características acústicas similares al segmento analizado.

El BK se obtiene por medio del siguiente procedimiento, que se ilustra en la Figura 17. En primer lugar, se seleccionan para cada vector de características acústicas de entrada (1) los índices de las N_G gaussianas del KBM (2) que mejor se ajustan a ellos (es decir, las N_G gaussianas con mayor similitud respecto al vector de entrada). Por cada segmento del que se han extraído características acústicas, se han identificado las N_G gaussianas del KBM que mejor se ajustan a ese segmento, y se establecen a 1 para indicar que son las gaussianas más importantes en ese segmento (3). El vector acumulativo (CV) v_c = {vc [1], …, v_c [N]}, con v_c [i] ϵ N, es un vector cuya posición i representa la gaussiana i en el KBM. Este vector, se inicializa a 0 en todas sus posiciones y la posición i del CV aumenta su valor en 1 en la posición i por cada segmento en el que esa gaussiana i ha sido identificada como una de las N_G gaussianas que mejor se ajustan, dando como resultado un vector CV que contiene la importancia de cada gaussiana del KBM para el vector de características acústicas correspondiente (4). De esta manera, quedan identificadas las gaussianas de mayor importancia para cada vector de características acústicas entrante.

El siguiente paso es obtener el vector v_f a partir del vector v_c. Para ello, se identifican los índices de las M posiciones de mayor valor de v_c y se establecen a 1 en v_f, manteniendo el resto de posiciones de v_f a 0 (5).

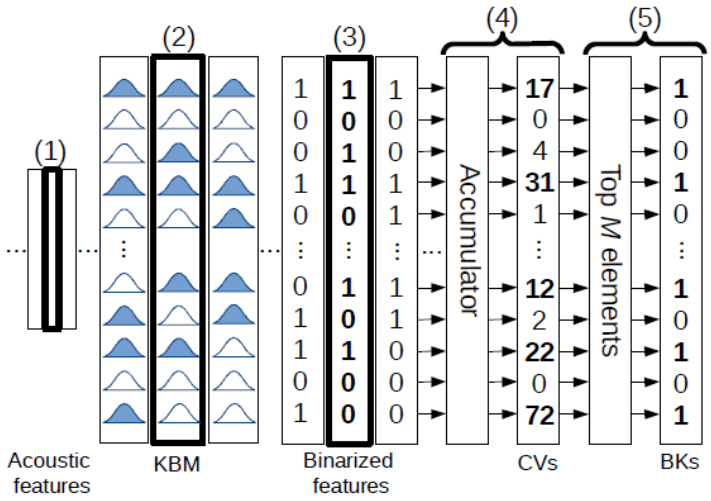


Figura 17: Ilustración de la extracción de los BK[21]

Una vez que las características acústicas de la señal de entrada se han convertido a BKs, las comparaciones se realizan de una manera más sencilla al reducirse la comparación únicamente a aplicar medidas de similitud entre vectores binarios.

Esta técnica requiere del cálculo de similitud de todas las características acústicas con todas las componentes gaussianas del KBM. Sin embargo, los valores de similitud son calculados una única vez y pueden ser guardados para utilizarlos en etapas posteriores, como en la parte de agrupamiento.

- ***Inicialización de los grupos***

La última etapa en el bloque de procesamiento acústico es establecer los grupos iniciales. Se utilizan las primeras N_{init} componentes del modelo KBM, obteniendo un agrupamiento sobre-segmentado de N_{init} grupos. De esta manera, se han seleccionado las N_{init} gaussianas de mayor diferencia entre ellas, que servirá como punto de partida para los grupos iniciales en la etapa de agrupamiento. Además, este método permite reutilizar las similitudes entre BKs obtenidas en el cálculo de los BKs. Este proceso de inicialización de los grupos se muestra en la Figura 18.

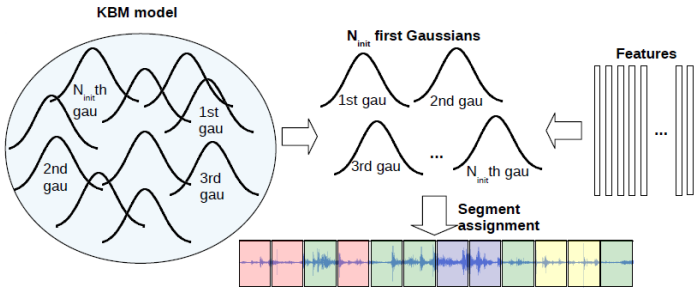


Figura 18: Inicialización del agrupamiento [4]

Módulo de Procesamiento Binario

En el módulo anterior, se han obtenido los BKs que representan las características acústicas de la señal de entrada y se han obtenido los grupos iniciales. El Módulo de Procesamiento Binario implementa un agrupamiento aglomerativo con la característica de que lo realiza utilizando la información binaria, por lo que el proceso es mucho más rápido que otros métodos como los basados en GMM.

El proceso comienza con N_{init} grupos. En cada iteración, se reasignan los BKs a los grupos actuales, y los dos grupos más similares se unen, reduciendo el número de grupos en una unidad. Este proceso se repite N_{init} iteraciones, hasta que exista un único grupo que contenga todos los segmentos de audio. Finalmente, se selecciona la solución óptima entre todas las soluciones intermedias del proceso.

En los siguientes apartados se describe el proceso de agrupamiento aglomerativo y la elección del número óptimo de grupos, que son las dos etapas que conforman el Módulo de Procesamiento Binario.

- ***Agrupamiento Aglomerativo***

Tras el Módulo de Procesamiento Acústico, los N_{init} grupos iniciales tienen sus BKs calculados. El proceso de agrupamiento aglomerativo sigue el procedimiento que se muestra en el esquema de la Figura 19.

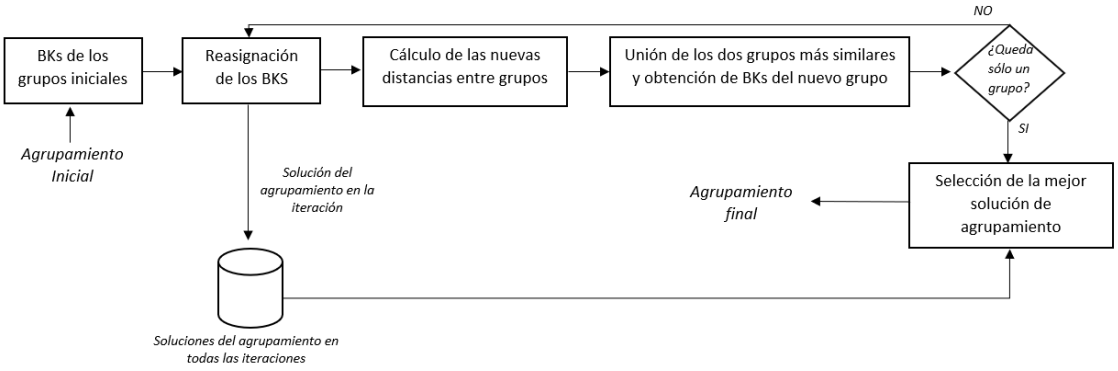


Figura 19: Esquema del agrupamiento aglomerativo. Imagen adaptada de [4]

1. Los BKs de entrada se reasignan a los grupos actuales. Este proceso se realiza calculando las similitudes entre cada BK de entrada y cada grupo, asignando los BKs al grupo que ofrece la mayor similitud, y calculando la medida de similitud de Jaccard, de acuerdo a la siguiente ecuación:

$$S(v_{f1}, v_{f2}) = \frac{\sum_{i=1}^{N}(v_{f1}[i] \wedge v_{f2}[i])}{\sum_{i=1}^{N}(v_{f1}[i] \vee v_{f2}[i])}$$

donde v_{f1} y v_{f2} son los BKs comparados, $\wedge$ es el operador booleano AND y $\vee$ el operador booleano OR. El valor de esta medida de similitud estará comprendido entre 0 y 1, con el

valor 0 indicando similitud nula (ninguno de los elementos coincide) y el valor 1 similitud total (todos los elementos son iguales).

2. Se calculan las distancias entre los BKs de cada grupo con la misma ecuación que en el paso anterior. El par de grupos con mayor similitud se unen en un único grupo, y se calcula el BK del nuevo grupo.
3. Se registra los valores de agrupamiento en esa iteración y el algoritmo vuelve al paso 1, mientras el número de grupos sea > 1. Si el número de grupos es igual a 1, se finaliza el proceso.

En cada iteración, se ha calculado una solución de agrupamiento, en la que se ha ido reduciendo en una unidad el número de grupos. Las soluciones obtenidas en cada iteración quedan guardadas para calcular posteriormente el número óptimo de grupos.

- **Selección del número óptimo de grupos**

En la etapa anterior, se han calculado N_{init} soluciones, una por cada iteración en la que el número de grupos descendía. La solución óptima es seleccionada entre todas las calculadas utilizando algún algoritmo de selección. El criterio empleado es una adaptación de la métrica T-test T_S descrita en [18]. Una solución de agrupamiento consiste en un número de grupos en el que cada grupo contiene elementos de características similares entre ellos y que tienen mayor diferencia respecto a elementos de los otros grupos, y están representados por BKs. Para el cálculo de T_s, primero se calculan las estadísticas de las distribuciones de similitud intra-grupo e inter-grupo (las distribuciones de todas las similitudes entre los BKs obtenidos en segmentos en el mismo grupo y entre todos los BKs de segmentos en los diferentes grupos). Después, suponiendo que ambas distribuciones son gaussianas, la medida T_s se calcula mediante la siguiente expresión:

$$Ts = \frac{m_1 - m_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$$

donde m_1, σ_1, n_1, m_2, σ_2 y n_2 son respectivamente la media, la desviación estándar y el tamaño de las distribuciones intra-grupo e inter-grupo. Finalmente, se escoge la solución de agrupamiento que maximice el valor de T_s

Mejoras sobre el sistema original

En los apartados anteriores se ha explicado el sistema original, describiendo sus dos bloques principales y los algoritmos y medidas utilizados en cada etapa. Sin embargo, tras evaluar el sistema con diferentes datos de entrada en varias evaluaciones, los autores del método comprobaron que era posible mejorar su rendimiento realizando algunas modificaciones. La validez de las modificaciones se ha comprobado experimentalmente en evaluaciones realizadas sobre una base de datos de ficheros de audio diferentes [19], en las que queda demostrado la mejora que suponen en el coste computacional del sistema completo sin afectar a los resultados. A continuación, se describen las mejoras realizadas en las diferentes etapas del sistema completo:

Entrenamiento de KBM

El principal "cuello de botella" del sistema completo es el proceso de entrenar el KBM, que penaliza el tiempo de ejecución total del sistema. Por tanto, conseguir reducir el tiempo de ejecución de este proceso supondrá una mejora importante en el rendimiento del sistema completo.

En el sistema original, la selección de las componentes gaussianas se realiza de manera iterativa calculando la distancia KL2 entre las gaussianas seleccionadas y las restantes. KL2 es una medida que indica cómo de diferentes son dos distribuciones de probabilidad. La divergencia DKL2 (Divergencia de Kullback - Leibler Simétrica) de dos distribuciones gaussianas P y Q se define de la siguiente manera:

$$D_{KL2}(P||Q) = D_{KL}(P||Q) + D_{KL}(Q||P)$$

donde DKL es la divergencia Kullback – Leibler de las distribuciones P y Q.

KL es una medida no-simétrica, y es por ello por lo que se emplea KL2. El cálculo de KL requiere de una serie de operaciones matriciales que hace que su coste computacional sea elevado.

Con el objetivo de encontrar un método más rápido y sencillo de implementar, se utiliza la siguiente mejora. Como se ha explicado anteriormente, el objetivo del proceso iterativo de selección de gaussianas es seleccionar las más discriminativas y complementarias. Como el cálculo de las medias (centro de la distribución) de las gaussianas puede considerarse suficiente para seleccionar las componentes más diferentes, se asume como criterio para la selección de las gaussianas. Por ello, se utiliza la **distancia coseno** entre los vectores de las medias de las gaussianas como medida de similitud. La distancia coseno *Dcos(a,b)* se define como:

$$Dcos(a,b) = 1 - Scos(a,b)$$

donde *Scos(a, b)* es la similitud coseno entre dos vectores a y b, definida como:

$$Scos(a,b) = \frac{a \cdot b}{\|a\| \, \|b\|}$$

La implementación de la distancia de coseno es mucho más sencilla que calcular KL2, por lo que se escoge esta medida en el proceso de selección de gaussianas.

Medidas de similitud en los Binary Keys

Los *binary keys* contienen la información acústica de la señal de entrada en vectores binarios. Las posiciones cuyo valor es igual a 1 en el BK indican que las gaussianas en esas posiciones en el KBM son las que mejor representan la secuencia de vectores de características entrantes. Los valores binarios de los BK se obtienen a partir del vector acumulativo (CV) que suma las veces que cada gaussiana del KBM es seleccionada, y cuyos valores más altos son convertidos a 1, como se ilustra en la Figura 16. El CV guarda el peso relativo de cada gaussiana del KBM respecto a un conjunto de características de entrada. Sin embargo, esa importancia, determinada por el valor que tiene cada gaussiana en el CV, se pierde en la conversión al BK, al cambiar a un valor binario. Por tanto, como esa información es útil para diferenciar los locutores, es posible hacer la comparación entre los CVs en lugar de los BKs.

Para realizar la comparación entre los diferentes CVs, es necesario establecer una medida de similitud, como la similitud de coseno *Scos*. Cada posición del CV contiene un entero positivo que representa cuantas veces la gaussiana del KBM asociada ha sido seleccionada entre las gaussianas principales. Por tanto, al ser los CVs vectores de enteros positivos, los valores de similitud resultantes estarán comprendidos en el intervalo (0, 1).

Selección de técnica de agrupamiento

El algoritmo Ts empleado en el sistema base para seleccionar el número óptimo de locutores, no ofrece buenos resultados, por lo que la búsqueda de otra técnica es fundamental para mejorar los resultados obtenidos por el sistema completo.

En la etapa de agrupamiento, en cada iteración se unen los dos grupos con mayor similitud en un único grupo y se recalculan las distancias entre los grupos independientes. Por cada iteración se obtiene una solución de agrupamiento, en la que el número de grupos independientes se reduce en una unidad respecto a la iteración anterior. La técnica propuesta utiliza la suma de cuadrados de las muestras dentro del grupo (*Within – Cluster Sum of Squares*, *WCSS*).

Siendo C_k la solución de agrupamiento compuesta por k grupos c_1, c_2, ..., c_k, la suma de cuadrados dentro del grupo (WCSS), $W(C_k)$, se define como:

$$W(C_k) = \sum_{i=1}^{k} \sum_{x \in c_i} \| x - \mu_i \|^2$$

donde μ_i es la media de los puntos del grupo c_i (es decir, el centro del grupo c_i). Cuanto menor sea WCSS, mejor será la solución.

El número óptimo de grupos se obtiene teniendo en cuenta la reducción de WCSS con cada iteración. Tras el proceso iterativo, se dispone de un conjunto de soluciones de agrupamiento C = (C_1, …, C_{Ninit}), cada una de ellas correspondiente a un número de grupos (desde un único clúster hasta N_{init} grupos) y WCSS es calculado para todas las soluciones. El máximo valor de WCSS se obtiene cuando N = 1, y a medida que N aumenta, la caída de WCSS es exponencial. En algún valor de N, la caída pasa de ser exponencial a prácticamente lineal. El primer punto que se desvía del comportamiento exponencial es el seleccionado y el número de grupos asociado es el que se determina como óptimo.

Como el punto donde la caída de los valores de WCSS pasa de exponencial a lineal no puede ser siempre determinado de una manera inequívoca, se ha propuesto una técnica para la identificación de este punto, que consiste en trazar la línea recta entre el valor de WCSS de la primera solución (N = 1) y la última (N = N_{init}) y calcular la distancia entre de cada punto de la curva y la línea recta. Tras el cálculo de las distancias, se selecciona el punto de mayor distancia a la línea recta y el número de grupos correspondientes a ese punto es el número óptimo de grupos. Este procedimiento se puede ver de manera gráfica en la Figura 20.

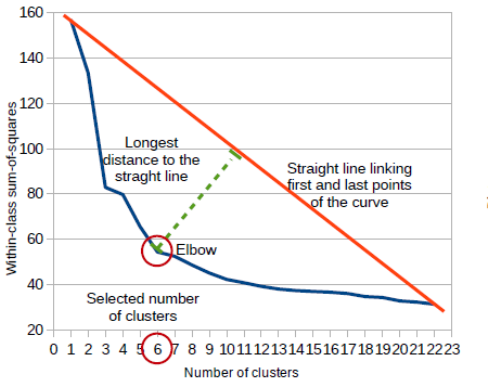


Figura 20: Ejemplo del criterio para obtener número óptimo de grupos en base a WCSS [19]

En el cálculo de WCSS, se pueden utilizar otras distancias en lugar de la distancia euclídea para el cálculo de las distancias entre cada punto del grupo y su media. Al emplear vectores acumulativos (CV) para representar los segmentos acústicos, es posible reemplazar la distancia euclídea por la distancia coseno, utilizada en etapas anteriores para comparar CVs. De esta manera, el valor de WCSS para una solución de agrupamiento C_k de k grupos $W(C_k)$ se puede reformular como:

$$W(C_k) = \sum_{i=1}^{k} \sum_{x \in c_i} (D(x, \mu_i))^2$$

siendo *D* la distancia coseno.

Mejoras recientes

Uno de los principales problemas para la evolución de los sistemas de diarización es la falta de *datasets* útiles sobre los que poder implementar los sistemas para poder evaluarlos. Recientemente, la iniciativa DIHARD ha desarrollado un *dataset* que contiene audios de muy diferentes características y que permite a los sistemas de diarización evaluar su funcionamiento sobre ellos [20]. Al mismo tiempo, el desarrollo de nuevas técnicas para la extracción de características y mejoras en técnicas de agrupamiento, han permitido aumentar el rendimiento de este sistema de diarización. La implementación del sistema con las mejoras recientes en el plan de evaluación DIHARD, ha ofrecido buenos resultados, obteniendo en su configuración con más mejoras un DER 29.33% de media tras evaluarlo en los ambientes acústicos del *dataset* [21].

A continuación, se describen las mejoras recientes aplicadas en el sistema para realizar el proceso de diarización de locutores.

Extracción de características acústicas

Una de las mejoras consiste en reemplazar los coeficientes MFCC por los coeficientes ICMC (*infinite impulse response, constant Q tansform Mel-frequency cepstral coefficients*), que se obtienen a partir del análisis espectro-temporal con la transformada Q [22], como alternativa a la transformada de Fourier a corto plazo (STFT). La principal diferencia entre ambos métodos es que la resolución espectro-temporal de STFT es constante, mientras que la Transformada Q Constante (CQT) tiene un factor Q fijo. El factor Q es una medida de la selectividad del filtro y se define como el cociente entre la frecuencia central y el ancho de banda. Por tanto, que el valor

de Q sea constante hace que haya mejor resolución espectral en bajas frecuencias y mejor resolución temporal en altas frecuencias. El principal inconveniente de esta técnica es su alto coste computacional.

Agrupamiento Espectral

Debido al éxito de realizar un agrupamiento espectral en otros sistemas de diarización, se implementa este agrupamiento con el modelo de *binary keys*. El agrupamiento espectral (AE) es un tipo de agrupamiento particional, es decir, no actúan de una manera jerárquica como el caso de AHC, sino que genera un conjunto de soluciones de agrupamiento dividiendo los datos en $\tilde{k}$ grupos. Este tipo de agrupamientos tienen la desventaja de que requieren de alguna técnica que permita estimar de manera explícita el número de locutores existentes para su aplicación en la diarización.

El agrupamiento espectral se fundamenta en el siguiente concepto: datos de la misma clase tiene como resultado una similitud alta y, por el contrario, datos de clases diferentes una similitud baja, y se podrán comparar al formar una matriz de afinidad (*affinity matrix*). En el caso concreto de utilizar el AE junto con los *binary keys*, consiste en el siguiente proceso: la matriz de afinidad se forma utilizando las similitudes obtenidas en los vectores acumulativos (CV), estimar los valores propios principales de la matriz y finalmente, obtener los vectores propios correspondientes, con los que se realiza el agrupamiento. Este procedimiento se puede dividir en las siguientes tres fases:

- *Pre – procesamiento*
  La primera fase consiste en aplicar operaciones de normalización a la matriz de afinidad para incrementar su homogeneidad. Estas operaciones se basan en la temporalidad local de la señal de voz, en la que segmentos contiguos de un mismo locutor tienen CVs similares y, por tanto, valores similares en la matriz de afinidad. Tras esta etapa de procesamiento, la matriz de afinidad permitirá diferenciar más fácilmente los grupos que tengan características diferentes.
  La matriz de afinidad formada a partir de BKs se define de la siguiente manera. El audio de entrada se representa por una secuencia de M CVs de dimensión N cada uno de ellos, siendo N la dimensión del KBM, formando una matriz J de dimensión MxN. La matriz de afinidad K, de dimensión MxM se determina usando la similitud del coseno como:

$$K = 1 - D_{cos}(J, J^T)$$

  Las operaciones realizadas sobre la matriz de afinidad se pueden ver en la Figura 21 y son los ajustes propuestos en [23]. Estas operaciones son las siguientes:
  Desenfoque gaussiano (Gaussian blurring): Aplicado con una desviación estándar σ, sirve para suavizar (*blur*) los datos de entrada, reduciendo inconsistencias producidas por el ruido.
  Descarte de similitudes por umbral (Row-wise thresholding): Los valores de una fila de la matriz que se encuentren por debajo del percentil *p* de esa fila se establecen a un valor de 0. Tras este paso, se eliminan valores que puedan relacionar a dos locutores diferentes.

Simetrización (Symmetrization): Permite recuperar la simetría perdida tras el paso anterior, que es de gran importancia para extraer los vectores propios sobre la matriz de afinidad final.

Difusión (Diffusion): Permite resaltar las fronteras entre secciones de la matriz de afinidad que pertenecen a locutores diferentes.

Normalización de fila (Row-wise max normalization): Consiste en una normalización final en la matriz para que sea más homogénea. Consiste en dividir cada elemento por el valor máximo de cada fila.

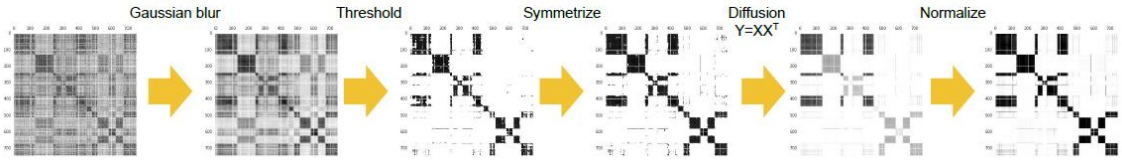


Figura 21: Operaciones realizadas sobre la matriz de afinidad [23]

- Número de grupos

  Una vez realizadas las operaciones anteriores sobre la matriz de afinidad, se extraen los vectores propios y sus respectivos valores propios, que se ordenan de manera descendente: $\lambda_1 > \lambda_2 > \ldots > \lambda_n$. El número de grupos $\tilde{k}$ se selecciona como el valor m que maximiza el ratio entre dos valores propios sucesivos (ratio conocido como *eigengap* o *separación propia*):

$$\tilde{k} = \arg \max_{1 \leq m \leq n} \frac{\lambda_m}{\lambda_{m+1}}$$

- Agrupamiento

  Tras determinar el número de grupos $\tilde{k}$ habiendo obtenido los vectores propios, el agrupamiento se puede realizar directamente en el dominio espectral. Los M vectores acumulativos de entrada se representan como una matriz de vectores propios de dimensión M x $\tilde{k}$. Se pueden emplear diferentes métodos para el agrupamiento, como puede ser K-means ya que el número de grupos se ha determinado. Otra alternativa sería emplear AHC, que es el método utilizado tanto en el sistema base como en el sistema mejorado, pero sin emplear criterio de parada, ya que el número de grupos es conocido.

Detección de audios con locutor único

Una última mejora introducida en el sistema es el diseño de un mecanismo que detecta que en el audio existe un único locutor, debido a que en este tipo de audios los errores de diarización producidos pueden ser elevados.

Este mecanismo consiste en calcular la diferencia entre los valores propios de mayor valor, λ_1 - λ_2 y si excede un umbral definido previamente, se determina que el número de grupos es 1, que corresponde a que en el audio de entrada hay un único locutor.

8.2.2 Diarización con SIDEKIT for Diarization (S4D)

Introducción a SIDEKIT y S4D

El segundo sistema de diarización utilizado en este trabajo ha sido la herramienta S4D[7] (*Sidekit for Diarization*), que es una extensión de la herramienta SIDEKIT[8]. SIDEKIT ofrece todas las herramientas necesarias para el reconocimiento de locutores: extracción de características (Coeficientes MFCC Y LFCC), modelado y clasificación (modelos GMM) e incluso técnicas para visualización de los resultados.

Al ser S4D una extensión de SIDEKIT, incluye las herramientas de SIDEKIT más las desarrolladas para realizar la diarización de locutores. Entre todas las técnicas adicionales que incluye S4D, las más importantes para este trabajo son la segmentación BIC y el agrupamiento jerárquico aglomerativo BIC.

Tanto SIDEKIT como S4D han sido desarrolladas en Python, y ofrecen la documentación completa sobre todas las clases y métodos disponibles, además de tutoriales que permiten al usuario poder implementar el sistema de manera sencilla.

S4D

La herramienta S4D, descrita en [24], realiza la diarización de locutores siguiendo el proceso tradicional de diarización (Ver Apartado de Estado del Arte): extracción de características, segmentación, agrupamiento y resegmentación. El usuario tiene a su disposición diferentes algoritmos en cada una de las etapas, por lo que puede elegir el que más se ajusta a las características de los audios a diarizar.

A continuación, se describen las utilidades que tiene S4D, donde se muestran las características principales en cada etapa del proceso de diarización, los parámetros más importantes en cada etapa y el formato utilizado para mostrar los resultados.

Extracción de características

La extracción de las características acústicas de la señal de audio de entrada se realiza mediante las clases *FeaturesExtractor* y *FeaturesServer* de SIDEKIT.

- *FeaturesExtractor*: procesa el fichero de audio de entrada y permite extraer los coeficientes cepstrales, definiendo el número de coeficientes, el tamaño y tipo de banco de filtros (lineal o logarítmico). También tiene la opción de aplicar el algoritmo de detección de actividad del habla (VAD), que en este trabajo no se aplica ya que se ha realizado anteriormente a la diarización.
- *FeaturesServer:* sirve la gestión de las características extraídas anteriormente. Permite un procesado posterior como puede ser la normalización o la selección de características. Se mantiene la configuración básica por defecto, ya que no se requiere un post-procesado sobre las características extraídas.

[7] Página web de S4D: https://projets-lium.univ-lemans.fr/s4d/
[8] Página web de SIDEKIT: https://projets-lium.univ-lemans.fr/sidekit/

Segmentación

Como se ha explicado anteriormente, la segmentación consiste en detectar los instantes en los que se produce un cambio de locutor. El algoritmo utilizado se basa en la detección de un máximo local, por medio de la divergencia gaussiana. Se calculan las gaussianas a la izquierda y derecha sobre una ventana que se va desplazando sobre la señal completa. El instante en la mitad de la ventana corresponde a un cambio de locutor si el valor de la divergencia gaussiana calculado entre las ventanas a izquierda y derecha alcanza un máximo local.

Tras la segmentación utilizando la divergencia gaussiana, se realiza una segunda operación sobre la señal completa que consiste en unir los segmentos consecutivos que corresponden a un mismo locutor, empleando el criterio ΔBIC. También permite usar la distancia basada en la raíz cuadrada BIC, definido en [25], para el valor de penalización en ΔBIC.

Agrupamiento

Para realizar el agrupamiento, S4D ofrece varios métodos. El principal es AHC BIC, que es el que se ha utilizado en la implementación del sistema. AHC BIC consiste en realizar el agrupamiento jerárquico aglomerativo (AHC), empleando el criterio BIC para juntar los grupos y como criterio de parada del proceso iterativo de agrupamiento. Cada grupo es modelado por una gaussiana y los dos grupos más similares i y j en cada iteración se unen hasta que se cumpla el criterio de parada: $\Delta BIC_{i,j} > 0$.

Los otros métodos disponibles para el agrupamiento son AHC CLR, que se basa en un agrupamiento jerárquico aglomerativo, pero con la medida CLR (*Cross Likelihood Ratio*) como medida de diferencia entre grupos y criterio de parada, ILP IV (*Integer Linear Programming I – Vector*) y AHC IV que extraen vectores – i de cada grupo para poder comparar las distancias entre ellos y determinar la similitud. Estos métodos se describen con mayor detalle en [24].

Resegmentación

Al igual que muchos otros sistemas de diarización de locutores, S4D incluye una última etapa en la que ajusta los límites entre locutores. Los diferentes locutores son representados por cada estado de un modelo HMM (*Hidden Markov* Model) y se aplica el algoritmo de Viterbi para detectar las fronteras de los segmentos de audio. Cada locutor es modelado por un GMM obtenido tras el algoritmo EM. En [3] se describe de forma más detallada el modelado HMM – GMM.

Formato de los resultados

Los resultados obtenidos tras la implementación de S4D tienen un formato específico, con un segmento en cada fila que representa cada turno de los diferentes locutores y que está compuesto por los siguientes 5 campos:

show cluster type start stop

donde *show* es el nombre del fichero, *cluster* es un identificador del grupo, *type* es el tipo de grupo, *start* es el instante de comienzo del turno (en centisegundos), y *stop* es el instante en el que finaliza el segmento (en centisegundos).

A pesar de que este formato es bastante intuitivo y sencillo de entender, debido a que tanto los ficheros de referencia empleados para la comparación como los resultados obtenidos en el sistema que utiliza *binary keys* tienen formato RTTM, se ha realizado la conversión de este formato a RTTM.

8.2.3 Evaluación de los sistemas

Se han descrito los dos métodos de diarización empleados en este trabajo, explicando tanto las características principales como las diferentes alternativas y mejoras en cada etapa. Para comprobar la adecuación de cada sistema con la BD utilizada y determinar con qué configuración se obtiene un mejor rendimiento, es importante evaluar los resultados obtenidos. En este apartado se describe el proceso seguido para evaluar los resultados obtenidos en cada sistema.

En primer lugar, se crean los ficheros RTTM de referencia, utilizados para determinar los intervalos de tiempo en los que habla cada locutor y poder compararlos con los obtenidos en los dos sistemas de diarización.

Después, se describen las características que deben reunir los ficheros de audio para ser utilizados como ficheros de prueba en los sistemas de diarización.

Finalmente, se muestra el criterio elegido para identificar al periodista y las consideraciones que se tienen en cuenta para la evaluación. Los resultados obtenidos en cada una de las diferentes configuraciones, permiten evaluar el rendimiento de cada sistema. La comparación de estos resultados, indican cuál de los métodos es el más adecuado para realizar la diarización de locutores en la BD de este trabajo.

Ficheros RTTM de referencia

Para poder evaluar los resultados obtenidos en los sistemas de diarización, es necesario disponer de ficheros de referencia que determinen qué locutor habla en cada momento. Estos ficheros tienen formato RTTM, y los ficheros con los resultados de la diarización obtenidos con cada sistema, que también tienen formato RTTM, son comparados con estos ficheros de referencia. Por tanto, cuantas más diferencias haya entre el fichero obtenido y los de referencia, mayor será el error de diarización (DER) cometido.

El primer paso es la identificación de los locutores en los ficheros de audio de manera manual. Para ello, se ha empleado la herramienta *wavesurfer*[9], que ofrece una gran variedad de funciones sobre los ficheros de audio, y especialmente útiles en señales de voz como la visualización del espectrograma. Otra de las funciones es poder marcar los cambios de locutor, determinando los intervalos de cada uno de ellos y exportarlos a un fichero *lab*, que contendrá por cada línea el identificador del locutor y los instantes de inicio y final del intervalo en el que habla. Con esa información, se realiza la conversión del fichero *lab* a fichero RTTM de manera sencilla, al contener los campos más importantes. En la Figura 22 se puede ver el proceso de marcado en *wavesurfer*, tras el que se obtienen los ficheros *lab*.

[9] Página web de wavesurfer: https://wavesurfer-js.org/

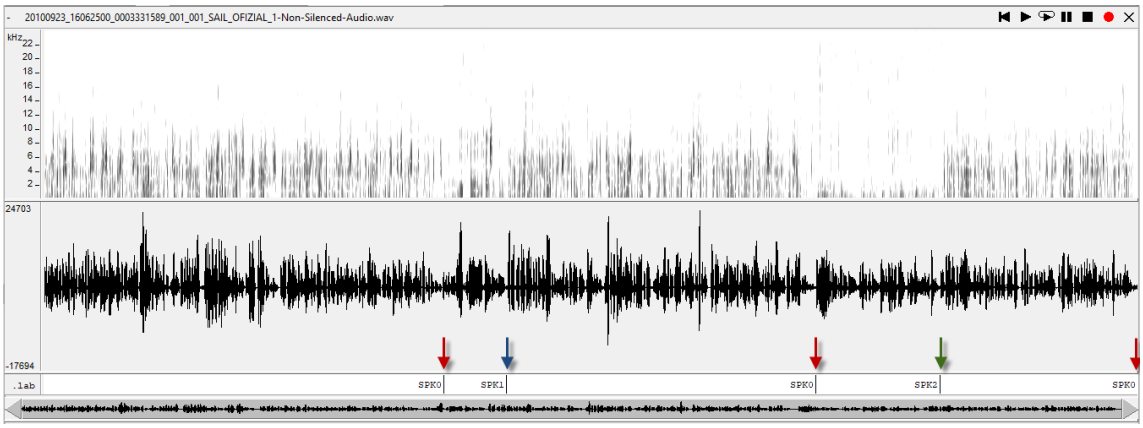


Figura 22: Marcado de fragmentos de cada locutor en wavesurfer

Selección de ficheros

Los sistemas de diarización se han evaluado utilizando un conjunto de ficheros extraídos de la base de datos. Los ficheros seleccionados se han escogido siguiendo estas consideraciones:

- **Audios sin ruidos o sonidos externos**. En general, todos los audios permiten escuchar las voces de los locutores en muy buenas condiciones, aunque existen algunos audios en los que se escuchan sonidos adicionales, como pueden ser aplausos o ruido de fondo en locutores que hablan en exteriores. Debido a las características de los audios, en aquellos en los que el número de personas entrevistadas es alto, existen algunos fragmentos en los que existe ruido de fondo al hablar en exteriores, pero en los que la voz de los locutores se puede escuchar con buena calidad.
- **Diversidad en cuanto al idioma y género de los locutores**. La selección contiene ficheros en los que la periodista principal es tanto mujer como hombre y el idioma empleado euskera y castellano.
- **Audios multilocutor**. Se han descartado de la selección aquellos ficheros en los que hay un único locutor.

En total, se han seleccionado **73** ficheros, en los que en 37 ficheros el locutor principal habla en euskera y 36 en castellano. Si se hace la clasificación por el género del locutor principal, en 38 ficheros la periodista es mujer y en 35 es hombre. El resumen de los ficheros por categorías se puede ver en la Tabla 6 y la información detallada sobre cada fichero se muestra en el Anexo 1.

	Euskera	**Castellano**
Mujer	19	19
Hombre	18	17

Tabla 6: Cantidad de ficheros seleccionados para la evaluación de los sistemas de diarización

Error en la detección del periodista

El DER determina el error cometido en la identificación de los locutores tras el proceso de diarización respecto a los locutores identificados en el fichero de referencia. Por tanto, es un error que considera de igual manera todos los locutores presentes en el audio de entrada. Sin embargo, el objetivo de este proyecto es identificar al periodista en los ficheros de audio, para

poder extraer únicamente las partes del audio que le corresponden. Por ello, resulta útil establecer una medida del error cometido en la identificación del locutor principal, sin considerar errores producidos en la identificación del resto de locutores.

El criterio elegido para identificar al periodista entre todos los locutores existentes en cada fichero de audio, ha sido seleccionar el locutor que más tiempo habla de todos ellos. En general, los ficheros corresponden a grabaciones en las que el periodista describe el contexto de la noticia al principio del audio, habla en uno o varios momentos intermedios para relacionar el contenido de diferentes locutores entrevistados y termina con una conclusión sobre la información descrita durante el audio. En el Anexo 1 se muestra una pequeña descripción de la secuencia de los audios utilizados en la selección de ficheros.

Con el objetivo de calcular el DER únicamente de la parte del periodista, una alternativa es modificar los ficheros RTTM tanto de referencia como los obtenidos tras los sistemas de diarización, extrayendo las partes que corresponden al resto de locutores. De esta manera, el DER resultante corresponde exclusivamente al periodista que se ha identificado.

El error producido en la diarización está formado por diferentes errores, como se muestra en el apartado 3.3. En el caso concreto del DER en RTTMs en los que únicamente contienen los instantes en los que habla el periodista, estará formado por los errores de *Missed Speaker* y *False Alarm*.

En el caso de *Missed Speaker*, el error es producido por los intervalos en los que en el RTTM del sistema no se detecta locutor y en el RTTM de referencia son instantes en los que el locutor habla. En el caso de *False Alarm*, el caso es el contrario, el sistema ha detectado que el locutor habla cuando en la referencia corresponde a silencio del locutor. En cuanto al error de identificar a un locutor incorrecto, *Speaker Error*, no influye en este caso concreto ya que únicamente se dispone de un único locutor en los ficheros RTTMs

Existe un *plug-in*[10] para *wavesurfer* que permite comprobar de manera visual los intervalos de los diferentes locutores tanto en el fichero RTTM de referencia como en el del sistema de diarización. De esta manera, se puede comprobar visualmente si se han detectado los locutores correctamente, como muestra la Figura 23.

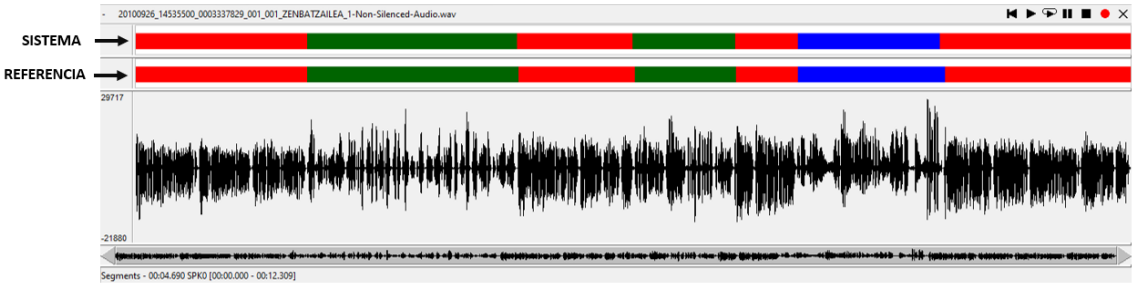


Figura 23: Visualización de los turnos de los locutores en los RTMM del Sistema/Referencia

[10] Página web del *plug-in*: http://www.xavieranguera.com/resources/resources.html

8.3 Implementación de los sistemas de diarización

En el apartado anterior se han descrito los dos sistemas de diarización de locutores, explicando los métodos utilizados en cada etapa, las posibles alternativas y criterios que se pueden emplear y en el caso del sistema que utiliza *binary keys*, las mejoras que se han introducido.

Ambos sistemas permiten modificar los parámetros de cada etapa del proceso de diarización, con el objetivo de obtener la configuración que mejor se adapte a cada caso. Los sistemas han sido evaluados utilizando diferentes configuraciones en evaluaciones anteriores, como en [19], [21] y [24]. En este trabajo, se va a analizar y evaluar el rendimiento de los dos sistemas sobre la base de datos de EITB, descrita en el apartado 8.1.

8.3.1 Sistema de diarización con binary keys

Este método ha ido añadiendo mejoras respecto a los métodos empleados en el sistema original. Por ello, se ha utilizado una de las configuraciones más recientes, con las mejoras incluidas: se emplea la distancia coseno para comparar los vectores acumulativos, se define un número grupos inicial y se utilizan nuevos métodos recientes para determinar el número de grupos para realizar el agrupamiento AHC. Se utilizan dos configuraciones diferentes, aplicando dos métodos distintos para determinar el número de grupos/locutores en la etapa de agrupamiento.

El primero de los métodos es el basado en la suma de los cuadrados del grupo (*WCSS*), que calcula el WCCS para todas las soluciones de agrupamiento y determina que el número óptimo de grupos es aquel que hace que la evolución del WCSS respecto al aumento de número de grupos pase de un descenso exponencial a un descenso lineal (Ver Figura 20). El segundo método es el agrupamiento espectral. Las dos configuraciones tienen las mismas características en el resto de las etapas. En la parametrización, se calculan 19 coeficientes MFCC cada 25 ms con un desplazamiento de ventana de 10 ms. El agrupamiento se realiza empleando la distancia de coseno para determinar la similitud entre los vectores acumulativos de cada grupo (mejora propuesta sobre el sistema original). Los parámetros utilizados en ambas configuraciones se resumen en las tablas Tabla 7 - Tabla 11, en la que se han abreviado *config_elbow* y *config_spectral.*

Del resto de las mejoras más recientes, no se implementarán los coeficientes ICMC para la extracción de características acústicas ni la detección de locutor único. La utilización de los coeficientes ICMC es descartada debido a su alto coste computacional respecto a los coeficientes MFCC, que además son los más utilizados en diarización. En cuanto a la detección de locutor único, al emplear audios en los que como mínimo hay dos locutores, no se ha optado por añadir esa detección.

- Extracción de características acústicas

	config_elbow	***config_spectral***
Número de coeficientes MFCC	19	19
Número de filtros de Mel	20	20
Tamaño de ventana extracción características	25 ms	25 ms
Desplazamiento entre ventanas	10 ms	10 ms

Tabla 7: Parámetros empleados en la extracción de características

- Binary Key Background Model (KBM)

	config_elbow	***config_spectral***
Mínimo número de gaussianas iniciales	1024	1024
Máxima frecuencia de ventanas para obtención de gaussianas iniciales	50	50
Tamaño de ventana para obtención de gaussianas	200	200
Porcentaje de gaussianas en el KBM	10%	10%

Tabla 8: Parámetros del KBM

Estos parámetros permiten definir el número de gaussianas iniciales y el número de gaussianas utilizadas en el KBM. El tamaño de ventana y la frecuencia de ventana determina cada cuántas ventanas en las que se divide la señal de audio de entradas se calculan las gaussianas, con el objetivo de tener como mínimo el número de gaussianas definido previamente.

Sobre el conjunto de gaussianas totales se obtienen las gaussianas más discriminativas para formar el KBM. El número de gaussianas que definen el KBM puede ser definido como un valor fijo previamente, o como un porcentaje respecto del número total de gaussianas. Al no tener todos los ficheros la misma duración, se ha optado definir un porcentaje respecto del total.

- Binary keys

	config_elbow	***config_spectral***
Número de gaussianas principales por trama	5	5
Porcentaje de bits a 1 en los BK	20%	20%

Tabla 9: Parámetros para la extracción de los Binary Keys

Se determina el número de gaussianas N_G del KBM en cuyas posiciones se suma una unidad al ser las que más se ajustan a las características acústicas de ese vector. El porcentaje de bits que se establece a 1 en el BK, determina cuántas posiciones del CV se establecen a 1 en el BK (los valores más altos del CV). Este proceso es el que se ha ilustrado en la Figura 16.

- Agrupamiento

	config_elbow	***config_spectral***
Número de grupos iniciales	16	16
Métrica para definir similitud	distancia coseno	distancia coseno

Tabla 10: Parámetros del agrupamiento

El número de grupos iniciales determina el número total de soluciones de agrupamiento sobre las que se obtiene el número óptimo de grupos. La métrica utilizada para determinar la similitud entre los grupos, es la distancia coseno, que tal y como se ha explicado anteriormente, es una medida sencilla de implementar que ofrece buenos resultados.

	config_elbow	***config_spectral***
Desviación estándar del desenfoque gaussiano (*Gaussian Blurring*, *σ*)	-	1
Percentil – p de las similitudes	-	40

Tabla 11: Parámetros del agrupamiento espectral

Los parámetros definidos para el agrupamiento espectral son los utilizados en el proceso explicado en la etapa de *pre-procesamiento,* en el que se realizan una serie de operaciones sobre la matriz de afinidad con el objetivo de poder determinar el número de grupos diferentes de manera más sencilla.

Resultados

De la Tabla 12 a la Tabla 15 se muestran los resultados obtenidos con ambas configuraciones *config_elbow* y *config_spectral* para cada uno de los ficheros de referencia seleccionados. Los resultados se han agrupado teniendo en cuenta el género del periodista y el idioma en que habla. El collar utilizado en las evaluaciones ha sido de 250 ms.

Periodista: Mujer – Castellano

	config_elbow				config_spectral			
Audio ID	DER (%)	DER Periodista (%)	Spks REF	Spks SIS	DER (%)	DER Periodista (%)	Spks REF	Spks SIS
Audio_001	11.23	0.00	4	3	11.23	0.00	4	3
Audio_002	0.00	0.00	2	3	0.01	0.00	2	2
Audio_003	0.20	0.33	2	2	0.20	0.33	2	2
Audio_004	0.00	0.00	2	2	0.00	0.00	2	2
Audio_005	11.75	0.07	2	3	12.63	1.49	2	4
Audio_006	0.90	0.00	4	4	12.72	19.09	4	3
Audio_007	0.01	0.02	3	3	3.87	0.38	3	4
Audio_008	0.07	0.17	5	5	16.37	0.17	5	4
Audio_009	0.13	0.21	4	4	12.40	19.91	4	3
Audio_010	0.12	0.20	2	2	0.12	0.20	2	2
Audio_011	4.57	1.49	2	3	2.87	3.50	2	2
Audio_012	16.14	0.63	12	4	10.41	0.79	12	7
Audio_013	5.60	0.47	4	5	5.60	0.47	4	5
Audio_014	6.17	0.00	4	3	6.17	0.00	4	3
Audio_015	6.42	3.22	2	3	2.84	5.11	2	2
Audio_016	9.64	25.96	6	4	52.23	129.25	6	2
Audio_017	0.13	0.16	2	2	0.13	0.16	2	2
Audio_018	13.69	14.59	7	3	13.69	14.59	7	3
Audio_019	6.77	0.00	3	4	6.77	0.00	3	4
DER Promedio	**4.92**	**2.5**	-	-	**8.96**	**10.29**	-	-

Tabla 12: Resultados del método binary keys. Periodista: Mujer - Castellano

Periodista: Hombre – Castellano

	config_elbow				config_spectral			
Audio ID	DER (%)	DER Periodista (%)	Spks REF	Spks SIS	DER (%)	DER Periodista (%)	Spks REF	Spks SIS
Audio_101	20.98	0.46	2	3	20.98	0.46	2	3
Audio_102	5.64	1.54	2	3	0.00	0.00	2	2
Audio_103	0.00	0.00	3	3	0.00	0.00	3	3
Audio_104	21.15	19.09	5	4	15.03	0.00	5	2
Audio_105	0.19	0.32	3	3	0.19	0.32	3	3
Audio_106	7.84	5.94	5	6	7.84	5.94	5	6
Audio_107	0.00	0.00	2	2	0.00	0.00	2	2
Audio_108	14.48	12.34	3	4	5.90	12.16	3	3
Audio_109	5.98	0.08	3	2	5.98	0.08	3	2
Audio_110	17.00	7.54	4	4	30.25	8.51	4	3
Audio_111	0.20	0.23	3	3	0.20	0.23	3	3
Audio_112	0.00	0.00	3	3	0.00	0.00	3	3
Audio_113	0.07	0.14	3	3	14.98	0.44	3	2
Audio_114	17.32	27.21	2	4	5.23	8.38	2	2
Audio_115	0.11	0.14	2	2	1.48	0.78	2	3
Audio_116	0.99	0.03	3	3	0.99	0.03	3	3
Audio_117	0.00	0.00	3	3	0.00	0.00	3	3
DER promedio	**6.59**	**4.42**	-	-	**6.41**	**2.2**	-	-

Tabla 13: Resultados del método binary keys. Periodista: Hombre - Castellano

Periodista: Mujer – Euskera

	config_elbow				config_spectral			
Audio ID	DER (%)	DER Periodista (%)	Spks REF	Spks SIS	DER (%)	DER Periodista (%)	Spks REF	Spks SIS
Audio_201	0.18	0.24	2	2	0.18	0.24	2	2
Audio_202	0.08	0.10	3	3	6.10	0.10	3	2
Audio_203	0.08	0.00	3	3	5.66	0.00	3	2
Audio_204	17.29	0.17	12	5	21.29	28.54	12	4
Audio_205	8.26	0.43	2	3	8.26	0.43	2	3
Audio_206	0.00	0.00	2	2	0.00	0.00	2	2
Audio_207	5.50	9.37	6	5	23.16	51.87	6	3
Audio_208	0.04	0.06	2	2	0.04	0.06	2	2
Audio_209	4.25	6.45	3	2	4.25	6.45	3	2
Audio_210	0.26	0.22	5	5	13.46	5.79	5	3
Audio_211	0.06	0.11	2	2	0.06	0.11	2	2
Audio_212	12.50	1.54	2	3	0.86	1.42	2	2
Audio_213	34.26	44.94	2	3	0.05	0.07	2	2
Audio_214	0.00	0.00	2	2	3.84	5.21	2	3
Audio_215	0.02	0.02	2	2	7.63	0.39	2	3
Audio_216	3.99	1.95	4	3	3.99	1.95	4	3
Audio_217	11.35	15.14	2	3	0.59	0.79	2	2
Audio_218	14.17	18.22	3	4	25.84	33.25	3	3
Audio_219	12.60	0.55	8	4	16.10	7.77	8	3
DER promedio	**6.57**	**5.24**	-	-	**7.44**	**7.6**	-	-

Tabla 14: Resultados del método binary keys. Periodista: Mujer - Euskera

Periodista: Hombre – Euskera

	config_elbow				config_spectral			
Audio ID	DER (%)	DER Periodista (%)	Spks REF	Spks SIS	DER (%)	DER Periodista (%)	Spks REF	Spks SIS
Audio_301	3.00	1.60	6	5	42.15	92.93	6	2
Audio_302	0.10	0.18	2	2	0.10	0.18	2	2
Audio_303	0.86	1.04	3	3	0.86	1.04	3	3
Audio_304	11.97	15.21	2	4	0.00	0.00	2	2
Audio_305	0.00	0.00	3	3	0.00	0.00	3	3
Audio_306	4.38	3.11	3	4	6.49	0.00	3	2
Audio_307	0.00	0.00	3	3	0.00	0.00	3	3
Audio_308	0.00	0.00	3	3	15.25	129.28	3	2
Audio_309	14.68	7.56	4	4	15.74	0.47	4	7
Audio_310	2.87	4.44	3	3	6.97	3.89	3	4
Audio_311	0.16	0.23	2	2	0.16	0.23	2	2
Audio_312	0.00	0.00	2	2	0.00	0.00	2	2
Audio_313	4.78	0.89	6	5	7.25	0.44	6	6
Audio_314	0.00	0.00	3	3	15.76	0.00	3	2
Audio_315	0.24	0.41	3	3	0.24	0.41	3	3
Audio_316	0.00	0.00	3	3	0.00	0.00	3	3
Audio_317	0.14	0.16	2	2	0.14	0.16	2	2
Audio_318	7.27	0.40	4	3	3.93	1.02	4	5
DER promedio	**2.8**	**1.96**	-	-	**6.39**	**12.78**	-	-

Tabla 15: Resultados del método binary keys. Periodista: Hombre - Euskera

A continuación, se resumen los resultados más importantes tras las evaluaciones, con el cálculo del porcentaje de acierto en la identificación del número exacto de locutores, y los errores promedio y la desviación estándar del DER.

Número de locutores determinado correctamente.

Config_elbow: Determina correctamente el número de locutores existentes en el fichero en 40/73, lo que supone el porcentaje de acierto en la detección ha sido de **54.8%**

Config_spectral: Determina correctamente el número de locutores existentes en el fichero en 36/73, lo que supone el porcentaje de acierto en la detección ha sido de **49.3%**

En la Figura 24 y Figura 25 se muestran los histogramas de las diferencias entre el número de locutores en el fichero de referencia y el determinado en ambas configuraciones. Se puede apreciar que, en el caso de no obtener el número correcto de locutores, la mayoría de las veces no se detecta el número de locutores por una diferencia de $\pm$ 1 locutor. Es decir, el sistema principalmente detecta un locutor de más o un locutor de menos.

En el caso del sistema con configuración *config_elbow (*Figura 24*)*, el sistema tiende a determinar que hay un locutor más de los que realmente hay (valor de diferencia REF - SIS: -1)

En el sistema con configuración *config_spectral (*Figura 25*)*, el sistema tiende a determinar que hay un locutor menos de los que realmente hay (valor de diferencia REF - SIS: +1)

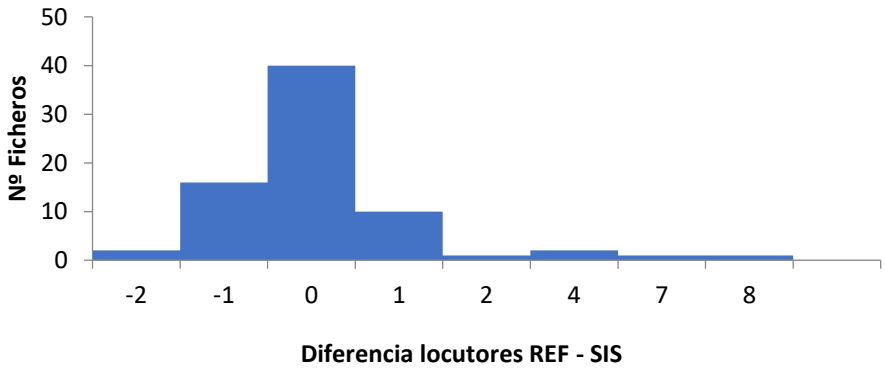


Figura 24: Histograma de la diferencia del número de locutores identificados (config_elbow)

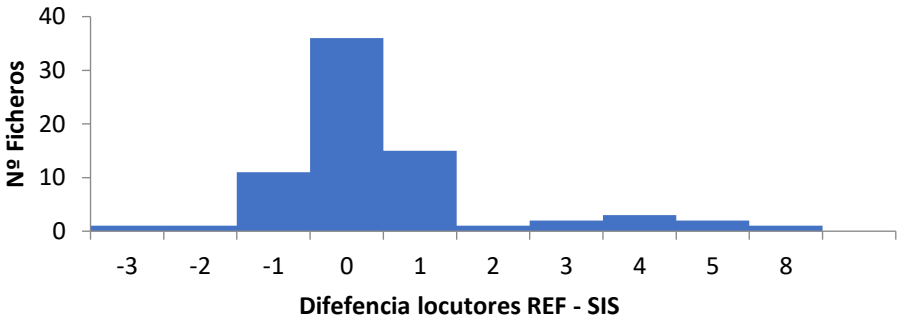


Figura 25: Histograma de la diferencia de número de locutores identificados(config_spectral)

Error en la Diarización (DER) de todos los locutores

Config_elbow: El error promedio cometido es de **5.22%,** con una desviación estándar de **7%**

Config_spectral: El error promedio cometido es de **7.34%,** con una desviación estándar de **9.92%**

Error en la Diarización (DER) únicamente del periodista

Config_elbow: El error promedio cometido es de **3.52%,** con una desviación estándar de **7.8%**

Config_spectral: El error promedio cometido es de **8.32%,** con una desviación estándar de **24.48%**

Tiempo de ejecución

Se ha medido el tiempo de ejecución total de ambas configuraciones. En ambas ejecuciones se ha utilizado el mismo sistema: Intel Core i5 – 8250U 1.6 GHz con RAM de 8 GB.

config_elbow: Tiempo de ejecución de 104 segundos para 73 ficheros de audio.

config_spectral: Tiempo de ejecución de 102 segundos para 73 ficheros de audio.

La elección del método para determinar el número óptimo de grupos en la etapa de agrupamiento no influye de manera significativa en el tiempo de ejecución. Un factor que influye directamente en el tiempo requerido para realizar la diarización es el número de gaussianas iniciales y el tamaño del KBM. En ambos casos valores mayores implican mayor procesamiento y, por tanto, mayor tiempo de ejecución total.

8.3.2 Sistema de diarización s4d

El sistema de diarización basado en SIDEKIT, permite utilizar varios métodos en cada etapa de la diarización, como se ha visto en el apartado 8.2.2. En el sistema, los parámetros que se han ajustado para las diferentes configuraciones son los siguientes:

- *Tamaño de la ventana en el cálculo de la divergencia gaussiana.* Determinada por la duración de las ventanas a izquierda y derecha del instante en el que se comprueba si se produce un cambio de locutor.
- *Umbral del criterio ΔBIC en la segmentación*. Permite definir lo estricta que sea la etapa de segmentación determinando que dos segmentos consecutivos pertenecen a un mimo locutor o no. Cuanto menor sea el umbral, implica mayor facilidad en que ΔBIC lo supere y, por tanto, se detecte como que los segmentos pertenecen a locutores diferentes.
- *Umbral del criterio ΔBIC en el agrupamiento jerárquico aglomerativo (AHC)*. Permite definir lo estricta que sea la etapa de agrupamiento determinando si dos grupos se unen en único grupo o no. Cuanto menor sea el umbral, implica mayor facilidad en que el valor de ΔBIC de la comparación lo supere y se determinen como grupos diferentes. Por tanto, cuanto menor sea el umbral, el número de grupos será mayor.
- *Umbral Viterbi*. Es un umbral utilizado en la etapa final de resegmentación, para ajustar los límites entre locutores. En las etapas de segmentación y agrupamiento previas a la resegmentación, se ha determinado el número de locutores existentes, por lo que esta etapa dispone de esa información. Este umbral determina la precisión al asignar cada intervalo de tiempo a cada locutor. Si no se usa un umbral adecuado, esta etapa determina muchos cambios de locutor de muy pequeña duración, lo que produce cortes intermedios en el turno de cada locutor.

Además, la implementación de este sistema permite modificar más parámetros de la diarización. Entre esos parámetros, el parámetro más importante ha sido determinar el número de coeficientes MFCC a 19, igual que en el sistema de diarización usando *binary keys*.

En primer lugar, se ha utilizado la configuración empleada en [24], con los parámetros que se muestran en la Tabla 16

Tamaño ventana segmentación – divergencia gaussiana	250 ms (izq) + 250 ms (der) = 500 ms
Umbral segmentación	2
Umbral agrupamiento	3
Umbral Viterbi	-250

Tabla 16: Parámetros de la configuración inicial de s4d en [24]

Utilizando esta configuración, los resultados muestran que estos parámetros no son los adecuados. El número de locutores existente se ha detectado correctamente en sólo 9 ficheros sobre los 73 de la selección.

El DER obtenido es de **28.41%**, con una desviación estándar de **15.74%.**

El DER con únicamente el periodista ha sido de **49.03%** y la desviación estándar de **36.7%.**

En el histograma de la Figura 26 se puede ver la distribución de la diferencia de locutores en el RTTM de referencia respecto al obtenido por el sistema. Se puede apreciar que, con esta configuración, todos los ficheros en los que no se ha detectado correctamente el número de locutores, ha sido debido a que el número de locutores existentes en el fichero de referencia es mayor que el detectado por el sistema.

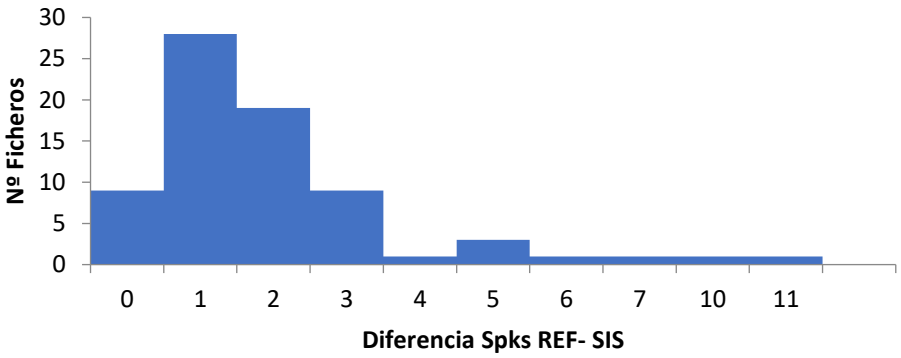


Figura 26: Histograma de diferencia del número de locutores con configuración inicial de s4d

El motivo de que la configuración por defecto no haya ofrecido buenos resultados en los ficheros seleccionados, es que sus parámetros fueron seleccionados para unos ficheros concretos, utilizados en las pruebas realizadas en [24]. Por tanto, el objetivo es encontrar los parámetros más adecuados y aplicarlos en la configuración del sistema para el caso concreto de la base de datos de este trabajo.

Análisis de los parámetros de configuración

Teniendo en cuenta el impacto de cada parámetro en el sistema, se han realizado experimentos con diferentes configuraciones, con el objetivo de encontrar aquellos parámetros más adecuados. A continuación, se describen las modificaciones realizadas en cada etapa y las consideraciones que se han tenido en cuenta para la elección de los nuevos parámetros. Para la nomenclatura de los parámetros, las abreviaturas son las siguientes: tamaño de ventana de segmentación: *win_size*; umbral en la segmentación: *thr_l*; umbral en el agrupamiento: *thr_h*; umbral de resegmentación: *thr_vit*.

La primera etapa en la que se ajustan parámetros es en la **segmentación**, en la que se establece el tamaño de ventana de las ventanas sobre las que se calculan las gaussianas. Por medio de la divergencia gaussiana y la posterior comparación de los segmentos usando el criterio ΔBIC, se determinan los cambios de locutor. Por tanto, un valor más pequeño de ventana hace que se divida la señal de audio en un mayor número de ventanas totales y un umbral *thr_l* más bajo permite que haya más cambios de locutor.

En la Figura 27 se muestra el resultado tras la segmentación con la configuración inicial en un fichero de audio seleccionado, en la Figura 28 tras reducir el umbral de segmentación y en la Figura 29 los resultados tras reducir tanto el umbral como el tamaño de ventana de segmentación. Cada línea mostrada en las figuras corresponde con un segmento detectado.

```
['audio', 'S0', 'speaker', 0, 876]
['audio', 'S2', 'speaker', 876, 7427]
['audio', 'S13', 'speaker', 7427, 8471]
['audio', 'S14', 'speaker', 8471, 9325]
```

Figura 27: Resultados segmentación s4d (thr_l = 2, win_size = 250) – Audio_301

```
['audio', 'S0', 'speaker', 0, 559]
['audio', 'S1', 'speaker', 559, 876]
['audio', 'S2', 'speaker', 876, 1390]
['audio', 'S3', 'speaker', 1390, 1816]
['audio', 'S4', 'speaker', 1816, 2808]
['audio', 'S6', 'speaker', 2808, 4999]
['audio', 'S9', 'speaker', 4999, 7008]
['audio', 'S12', 'speaker', 7008, 7427]
['audio', 'S13', 'speaker', 7427, 8471]
['audio', 'S14', 'speaker', 8471, 9325]
```

Figura 28: Resultados segmentación s4d (thr_l = 1, win_size = 250) – Audio_301

```
['audio', 'S0', 'speaker', 0, 626]
['audio', 'S2', 'speaker', 626, 868]
['audio', 'S3', 'speaker', 868, 1141]
['audio', 'S4', 'speaker', 1141, 1302]
['audio', 'S5', 'speaker', 1302, 1816]
['audio', 'S6', 'speaker', 1816, 2808]
['audio', 'S9', 'speaker', 2808, 3941]
['audio', 'S14', 'speaker', 3941, 4154]
['audio', 'S15', 'speaker', 4154, 4943]
['audio', 'S17', 'speaker', 4943, 7065]
['audio', 'S23', 'speaker', 7065, 7427]
['audio', 'S24', 'speaker', 7427, 8161]
['audio', 'S27', 'speaker', 8161, 8450]
['audio', 'S28', 'speaker', 8450, 9325]
```

Figura 29: Resultados segmentación s4d (thr_l = 1, win_size = 125) – Audio_301

Como se puede comprobar, un valor menor en ambos parámetros resulta en un mayor número de segmentos, que es mejor para poder detectar correctamente fragmentos cortos de los locutores. Además, la etapa de segmentación es de gran importancia ya que el agrupamiento se realiza con los resultados de esta etapa. Si en la segmentación se detecta como un único locutor intervalos en los que hay voz de dos o más locutores, no se recuperará esa información.

En cuanto al **agrupamiento**, un umbral menor favorece que en la comparación de los dos grupos para los que se ha calculado ΔBIC se determinen como grupos diferentes. Por el contrario, un valor mayor del umbral, implica que en la comparación entre los dos grupos haya mayor probabilidad de que se detecten como un mismo grupo. En la Figura 30 se muestran los resultados en la diarización de un fichero tras el agrupamiento con un valor de umbral de 2 y en la Figura 31 con un umbral de 3. En ambos casos, los resultados obtenidos tras la etapa previa de segmentación han sido los mismos.

```
['audio', 'S0', 'speaker', 0, 530]
['audio', 'S2', 'speaker', 530, 1100]
['audio', 'S0', 'speaker', 1100, 1638]
['audio', 'S2', 'speaker', 1638, 2164]
['audio', 'S2', 'speaker', 2164, 2312]
['audio', 'S2', 'speaker', 2312, 2642]
['audio', 'S0', 'speaker', 2642, 3092]
['audio', 'S10', 'speaker', 3092, 4070]
['audio', 'S0', 'speaker', 4070, 4530]
['audio', 'S13', 'speaker', 4530, 4965]
['audio', 'S14', 'speaker', 4965, 5657]
['audio', 'S0', 'speaker', 5657, 6213]
```

Figura 30: Resultados agrupamiento s4d (thr_l = 1, win_size = 125, thr_h = 2) – Audio_016

```
['audio', 'S0', 'speaker', 0, 530]
['audio', 'S13', 'speaker', 530, 1100]
['audio', 'S0', 'speaker', 1100, 1638]
['audio', 'S13', 'speaker', 1638, 2164]
['audio', 'S13', 'speaker', 2164, 2312]
['audio', 'S13', 'speaker', 2312, 2642]
['audio', 'S0', 'speaker', 2642, 3092]
['audio', 'S10', 'speaker', 3092, 4070]
['audio', 'S0', 'speaker', 4070, 4530]
['audio', 'S13', 'speaker', 4530, 4965]
['audio', 'S0', 'speaker', 4965, 5657]
['audio', 'S0', 'speaker', 5657, 6213]
```

Figura 31: Resultados agrupamiento s4d (thr_l = 1, win_size = 125, thr_h = 3) – Audio_016

La elección de este umbral es complicada, ya que no existe un valor que ofrezca los mejores resultados en todos los ficheros. En función de las características de cada fichero, principalmente del número de locutores, el umbral óptimo será diferente.

Una manera más visual de ver la formación de los grupos resultantes en el agrupamiento es por medio de un **dendrograma**. En el proceso de agrupamiento, en cada etapa se unen los dos grupos que tengan mayor similitud en un único grupo y se calculan las distancias del nuevo grupo respecto al resto de grupos. En el dendrograma, se puede comprobar visualmente el orden seguido en el agrupamiento de los grupos para formar un único grupo y el número de grupos independientes resultantes tras el proceso.

En la Figura 32 y la Figura 33 se pueden ver dos ejemplos de dendrogramas. En primer lugar, se calculan las distancias entre grupos en base a alguna medida. En este caso, se calcula ΔBIC entre todos los diferentes grupos y mientras no se cumpla el criterio de parada, se sigue el proceso de agrupamiento. Cuanto más bajo sea el valor en el eje vertical, que indica la distancia, implica que hay mayor similitud entre grupos y que, por tanto, en cada etapa los dos grupos que menor valor de distancia tengan se unen.

En el agrupamiento que se ilustra por medio de los dendrogramas de la Figura 32 y la Figura 33 los dos primeros grupos en unirse son “S0” Y “S18”. Tras unir esos dos grupos, la menor distancia es la que hay entre el nuevo grupo formado y “S13”, por lo que se son los siguientes grupos en unirse, dando como resultado un único grupo. El proceso se repite hasta que todas las distancias

entre los grupos sean mayores que 0 (línea roja horizontal). En esa última iteración, los grupos independientes (los que queden por debajo de la línea roja), son los grupos resultantes tras la etapa de agrupamiento. Los dendrogramas utilizados como ejemplo corresponden a dos configuraciones diferentes: una primera (Figura 32), en la que *thr_h* = 3 y en la que se identifican 4 locutores (*"S0", "S10", "S15", "S19")* y una segunda (Figura 33), utilizando *thr_h* = 2, que da como resultado 6 locutores diferentes (*"S0", "S9", "S10", "S15", "S19", "S22"*).

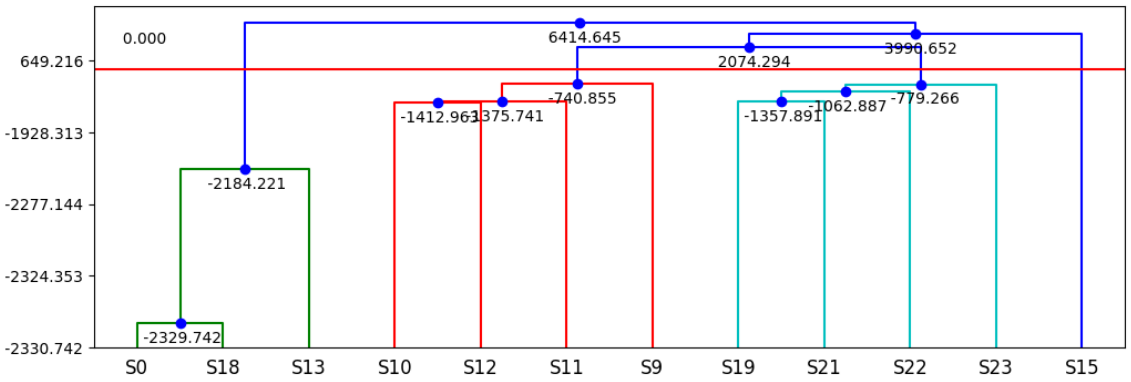


Figura 32: Resultados agrupamiento s4d (thr_l = 1, win_size = 125, thr_h = 3) – Audio_013

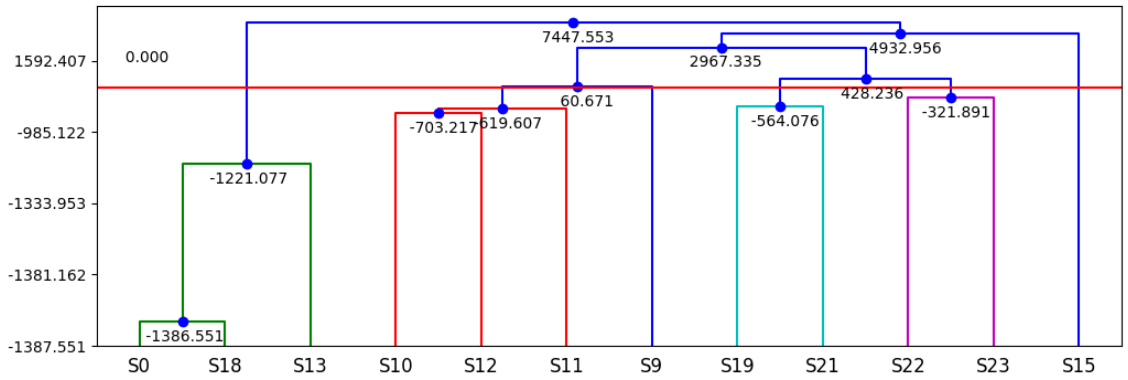


Figura 33: Resultados agrupamiento s4d (thr_l = 1, win_size = 125, thr_h = 2) – Audio_013

Por último, en la **resegmentación**, el valor del umbral se ha establecido igual que el utilizado en la configuración original, debido a que es el que mejores resultados ha ofrecido.

En resumen, se ha optado por reducir el umbral en la segmentación y el tamaño de ventana de segmentación para obtener más segmentos y evitar no detectar locutores desde el principio y probar diferentes valores en el agrupamiento. Las configuraciones utilizadas se muestran en la Tabla 17.

	window_size (ms)	*thr_l*	*thr_h*	*thr_vit*
config_1	125	1	3	-250
config_2	125	1	2	-250
config_3	125	1	2.5	-250

Tabla 17: Configuraciones utilizadas en sistema de diarización s4d

Las configuraciones utilizan un distinto umbral en el agrupamiento, que determinará que el sistema completo detecte más o menos locutores en el audio. A pesar de que se haya utilizado valores diferentes en el agrupamiento, en varios ficheros se han obtenido los mismos resultados. Esto se debe a haber utilizado el mismo umbral y tamaño de ventana en la segmentación.

Resultados

De la Tabla 18 a la Tabla 21 se muestran los resultados obtenidos con las configuraciones *config_1, config_2 y config_3* para cada uno de los ficheros de referencia seleccionados. Los resultados se han agrupado teniendo en cuenta el género del periodista y el idioma en que habla. El collar utilizado en las evaluaciones ha sido de 250 ms.

Periodista: Mujer – Castellano

	config_1				config_2				config_3			
	DER (%)	DER_Periodista (%)	Spks REF	Spks SIS	DER	DER_Periodista (%)	Spks REF (%)	Spks SIS	DER (%)	DER_Periodista (%)	Spks REF	Spks SIS
Audio_001	11.16	0.00	4	3	3.52	5.69	4	5	0.00	0.00	4	4
Audio_002	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_003	0.45	0.72	2	2	0.45	0.72	2	2	0.45	0.72	2	2
Audio_004	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_005	0.42	0.67	2	2	11.93	0.36	2	3	11.93	0.36	2	3
Audio_006	10.16	0.00	4	3	4.18	2.00	4	5	10.16	0.00	4	3
Audio_007	0.00	0.00	3	3	0.00	0.00	3	3	0.00	0.00	3	3
Audio_008	5.27	0.00	5	4	0.03	0.07	5	5	5.27	0.00	5	4
Audio_009	8.95	0.00	4	3	0.31	0.00	4	4	8.95	0.00	4	3
Audio_010	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_011	18.67	22.96	2	1	18.67	22.96	2	1	18.67	22.96	2	1
Audio_012	23.92	30.67	12	2	18.58	16.77	12	6	23.91	30.87	12	2
Audio_013	0.06	0.12	4	4	7.90	0.12	4	6	0.06	0.12	4	4
Audio_014	36.53	63.60	4	1	36.53	63.60	4	1	36.53	63.60	4	1
Audio_015	0.52	0.93	2	2	0.52	0.93	2	2	0.52	0.93	2	2
Audio_016	16.32	26.06	6	3	3.52	0.00	6	5	3.52	0.00	6	5
Audio_017	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_018	22.46	27.96	7	2	8.07	0.00	7	4	13.69	17.38	7	3
Audio_019	11.01	18.96	3	3	0.73	1.26	3	3	11.01	18.96	3	3
DER promedio	**8.73**	**10,14**	-	-	**6,05**	**6,03**	-	-	**7,61**	**8,21**	-	-

Tabla 18: Resultados del método s4d. Periodista: Mujer - Castellano

Periodista: Hombre – Castellano

	config_1				config_2				config_3			
	DER (%)	DER_Periodista (%)	Spks REF	Spks SIS	DER (%)	DER_Periodista (%)	Spks REF	Spks SIS	DER (%)	DER_Periodista (%)	Spks REF	Spks SIS
Audio_101	0.29	0.52	2	2	21.02	0.52	2	3	0.29	0.52	2	2
Audio_102	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_103	0.00	0.00	3	3	0.00	0.00	3	3	0.00	0.00	3	3
Audio_104	15.03	0.00	5	2	8.87	0.00	5	3	8.87	0.00	5	3
Audio_105	19.56	2.69	3	2	1.64	2.69	3	3	1.64	2.69	3	3
Audio_106	31.70	36.88	5	3	18.18	39.17	5	5	26.98	37.02	5	4
Audio_107	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_108	7.33	15.10	3	3	15.82	15.10	3	4	7.33	15.10	3	3
Audio_109	5.93	0.00	3	2	5.93	0.00	3	2	5.93	0.00	3	2
Audio_110	17.28	8.88	4	4	17.28	8.88	4	4	17.28	8.88	4	4
Audio_111	5.08	6.05	3	2	5.08	0.00	3	2	5.08	0.00	3	2
Audio_112	0.12	0.25	3	3	0.12	0.25	3	3	0.12	0.25	3	3
Audio_113	0.25	0.46	3	3	11.84	21.49	3	4	11.84	21.49	3	4
Audio_114	0.00	0.00	2	2	6.42	10.28	2	3	0.00	0.00	2	2
Audio_115	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_116	10.32	0.21	3	2	0.06	0.10	3	3	10.32	0.21	3	2
Audio_117	0.00	0.00	3	3	0.00	0.00	3	3	0.00	0.00	3	3
DER Periodista	**6.64**	**4.18**	-	-	**6.60**	**5.79**	-	-	**5.63**	**5.07**	-	-

Tabla 19: Resultados del método s4d. Periodista: Hombre - Castellano

Periodista: Mujer - Euskera

	config_1				config_2				config_3			
	DER (%)	DER_Periodista (%)	Spks REF	Spks SIS	DER	DER_Periodista (%)	Spks REF	Spks SIS	DER	DER_Periodista (%)	Spks REF	Spks SIS
Audio_201	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_202	12.89	8.94	3	3	6.87	8.94	3	4	6.87	8.94	3	4
Audio_203	5.64	115.16	3	2	0.41	0.00	3	3	0.41	0.00	3	3
Audio_204	24.64	21.58	12	3	13.38	3.59	12	6	20.84	3.30	12	4
Audio_205	19.50	24.22	2	1	8.98	11.16	2	2	19.50	24.22	2	1
Audio_206	0.02	0.04	2	2	0.02	0.04	2	2	0.02	0.04	2	2
Audio_207	7.98	20.71	6	4	7.98	20.71	6	4	7.98	20.71	6	4
Audio_208	0.18	0.24	2	2	0.18	0.24	2	2	0.18	0.24	2	2
Audio_209	4.25	6.45	3	2	4.25	6.45	3	2	4.25	6.45	3	2
Audio_210	20.18	18.52	5	2	10.37	6.15	5	5	11.64	12.78	5	3
Audio_211	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_212	23.24	39.61	2	2	22.08	12.05	2	3	23.24	39.61	2	2
Audio_213	0.18	0.24	2	2	0.18	0.24	2	2	0.18	0.24	2	2
Audio_214	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_215	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_216	5.98	10.17	4	3	5.98	10.17	4	3	5.98	10.17	4	3
Audio_217	2.64	3.53	2	2	2.64	3.53	2	2	2.64	3.53	2	2
Audio_218	11.95	15.38	3	2	13.73	17.66	3	4	13.73	17.66	3	3
Audio_219	19.39	15.36	8	2	15.31	0.00	8	3	19.39	15.36	8	2
DER Promedio	**8.35**	**15.8**	-	-	**5.91**	**5.31**	-	-	**7.2**	**8.59**	-	-

Tabla 20: Resultados del método s4d. Periodista: Mujer - Euskera

Periodista: Hombre – Euskera

	config_1				config_2				config_3			
	DER	DER_Periodista	Spks REF	Spks SIS	DER	DER_Periodista	Spks REF	Spks SIS	DER	DER_Periodista	Spks REF	Spks SIS
Audio_301	7.87	0.94	6	3	7.87	0.94	6	3	7.87	0.94	6	3
Audio_302	0.19	0.34	2	2	0.19	0.34	2	2	0.19	0.34	2	2
Audio_303	17.85	21.73	3	1	12.85	10.55	3	2	10.55	12.85	3	2
Audio_304	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_305	13.54	0.00	3	2	0.00	0.00	3	3	0.00	0.00	3	3
Audio_306	6.49	0.00	3	2	6.49	0.00	3	2	6.49	0.00	3	2
Audio_307	0.00	0.00	3	3	0.00	0.00	3	3	0.00	0.00	3	3
Audio_308	15.25	129.28	3	2	15.25	129.28	3	2	15.25	129.28	3	2
Audio_309	6.88	12.16	4	3	23.04	33.67	4	5	23.04	33.67	4	5
Audio_310	35.36	54.70	3	1	19.30	29.85	3	2	35.36	54.70	3	1
Audio_311	0.39	0.56	2	2	0.39	0.56	2	2	0.39	0.56	2	2
Audio_312	0.00	0.00	2	2	0.00	0.00	2	2	0.00	0.00	2	2
Audio_313	18.54	2.33	6	3	11.55	2.67	6	4	11.55	2.67	6	4
Audio_314	0.20	0.36	3	3	0.20	0.36	3	3	0.20	0.36	3	3
Audio_315	14.64	0.00	3	2	0.00	0.00	3	3	0.00	0.00	3	3
Audio_316	14.31	24.55	3	2	0.00	0.00	3	3	0.00	0.00	3	3
Audio_317	9.72	10.76	2	1	0.00	0.00	2	2	0.00	0.00	2	2
Audio_318	12.00	0.00	4	2	3.16	0.00	4	4	6.95	0.00	4	3
DER Periodista	**9.62**	**14.32**	-	-	**5.57**	**11.57**	-	-	**6.55**	**13.08**	-	-

Tabla 21: Resultados del método s4d. Periodista: Hombre - Euskera

Número de locutores determinado correctamente.

config_1: Determina correctamente el número de locutores existentes en el fichero en 37/73, lo que supone el porcentaje de acierto en la detección ha sido de **50.68%**

config_2: Determina correctamente el número de locutores existentes en el fichero en 42/73, lo que supone el porcentaje de acierto en la detección ha sido de **57.53%**

config_3: Determina correctamente el número de locutores existentes en el fichero en 42/73, lo que supone el porcentaje de acierto en la detección ha sido de **57.53%**

En Figura 34, Figura 35 y Figura 36 se muestran los histogramas de las diferencias entre el número de locutores en el fichero de referencia y el determinado en las configuraciones.

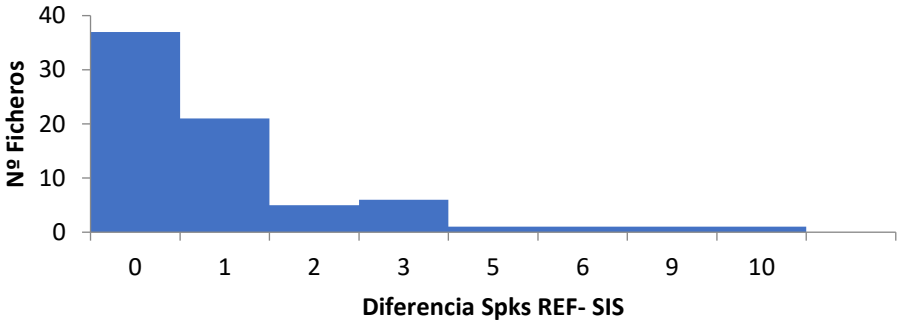


Figura 34:Histograma de la diferencia del número de locutores identificados (config_1)

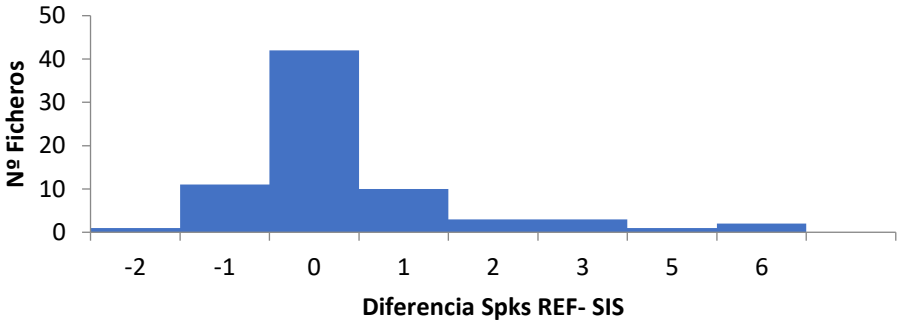


Figura 35: Histograma de la diferencia del número de locutores identificados (config_2)

-

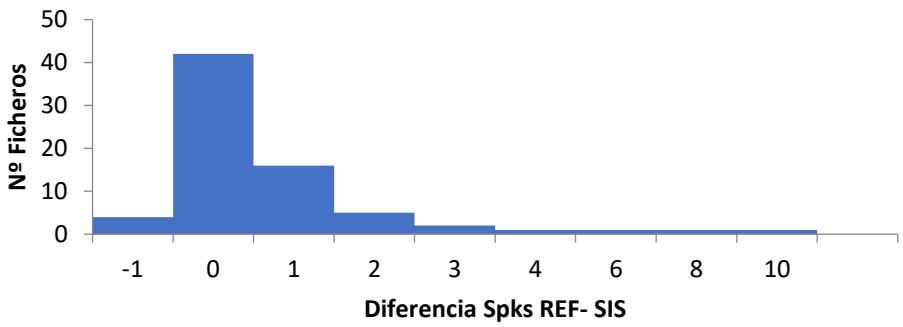


Figura 36:Histograma de la diferencia del número de locutores identificados (config_3)

En los histogramas se puede apreciar el efecto que tiene cambiar el valor del parámetro *thr_h* en el agrupamiento. El valor más alto de *thr_h* que se ha utilizado (*thr_h* = 3), es con el que menos grupos independientes se obtienen tras el agrupamiento. Por tanto, al aumentar el valor de *thr_h* el sistema tiende a detectar menos locutores que los que existen realmente, lo que se puede apreciar en la Figura 34, correspondiente a la configuración *config_1* y en la que en todos los audios en los que no se ha detectado el número de locutores correctamente, ha sido porque el sistema ha detectado un número menor (REF –SIS > 0).

Por otro lado, la configuración *config_2*, que es la que menor valor tiene de *thr_h* (*thr_h* = 2), ha detectado menos veces un número menor de locutores que los existentes (Figura 35), pero ha detectado en más ocasiones un número mayor de locutores (REF – SIS < 0).

Error en la Diarización (DER) de todos los locutores

config_1: El error promedio cometido es **8.37%,** con una desviación estándar de **9.44%**

config_2: El error promedio cometido es de **6.03%,** con una desviación estándar de **7.76%**

config_3: El error promedio cometido es de **6.78%,** con una desviación estándar de **8.88%**

Error en la Diarización (DER) únicamente del periodista

config_1: El error promedio cometido es de **11.25%,** con una desviación estándar de **23.04%**

config_2: El error promedio cometido es de **7.15%,** con una desviación estándar de **18.16%**

config_3: El error promedio cometido es de **8.78%,** con una desviación estándar de **19.46%**

Tiempo de ejecución

Se ha medido el tiempo de ejecución total del proceso con las diferentes configuraciones. En las tres ejecuciones se ha utilizado el mismo sistema: Intel Core i5 – 8250U 1.6 GHz con RAM de 8 GB.

config_1: Tiempo de ejecución de 88 segundos para 73 ficheros de audio.

config_2: Tiempo de ejecución de 87 segundos para 73 ficheros de audio.

config:3: Tiempo de ejecución de 87 segundos para 73 ficheros de audio.

9. Análisis de resultados

Tras el análisis de los resultados obtenidos con diferentes sistemas y configuraciones, el sistema que mejores resultados ha ofrecido ha sido el sistema que utiliza *binary keys* con la configuración *config_elbow*. Este sistema con esa configuración ha sido con el que menor DER se ha obtenido tanto en la evaluación de todos los locutores como en la evaluación únicamente del periodista.

La comparativa de las diferentes configuraciones analizadas se puede ver en la Figura 37 y la Figura 38. Se puede apreciar que los valores de DER evaluando todos los locutores ofrece valores similares en todas las configuraciones (Figura 37), sin que un sistema sea claramente mejor. En cuanto al DER producido en la detección del periodista (Figura 38), el DER ha sido menor utilizando config_elbow, con un margen ligeramente mayor que en el caso de DER general, aunque sin existir una diferencia importante entre sistemas.

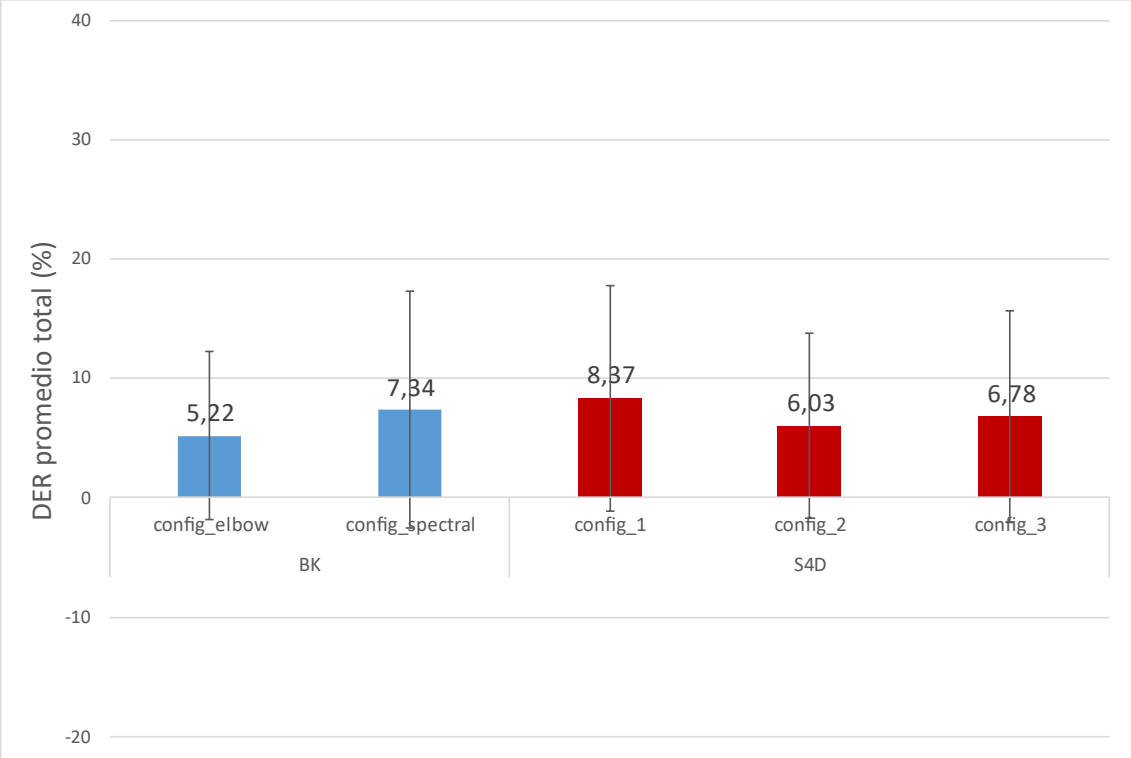


Figura 37: DER promedio total

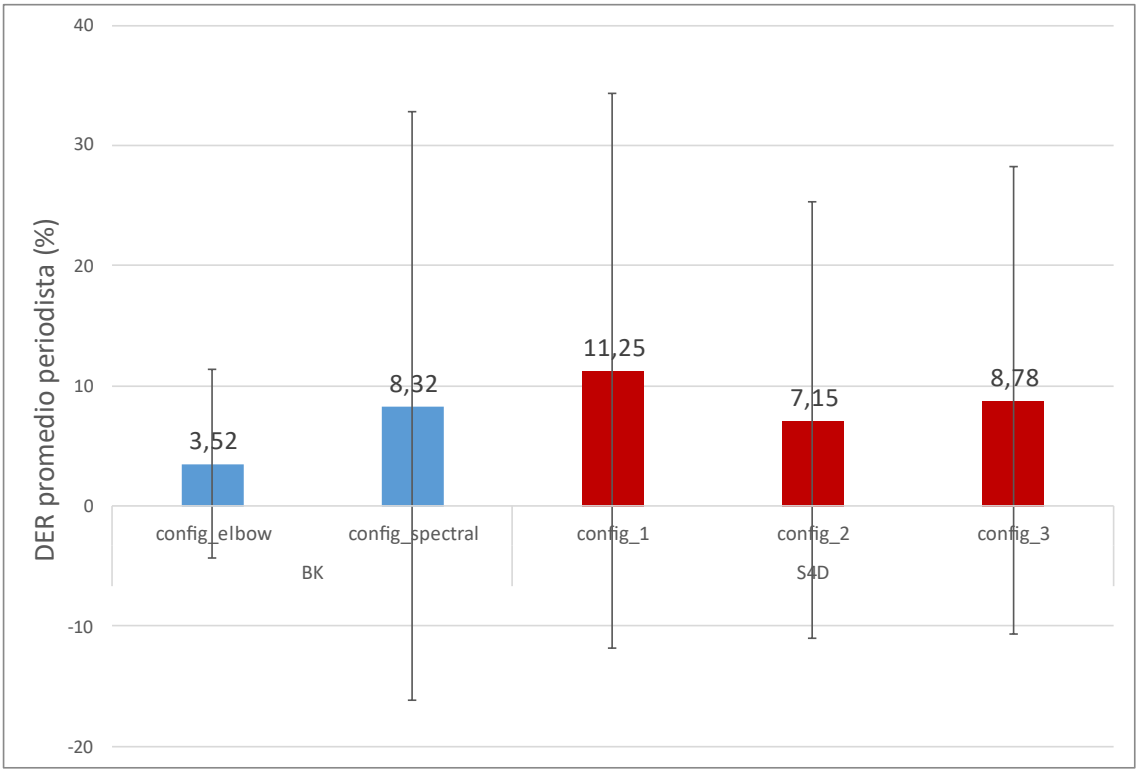


Figura 38: DER promedio periodista

Sin embargo, se ha observado que la desviación estándar de los resultados de DER obtenidos es alta, principalmente en el DER del periodista. Esto puede deberse a varios motivos como a las características del audio de entrada o a ficheros en los que no se ha detectado correctamente el número de locutores y, por tanto, los intervalos del periodista no se han podido detectar correctamente. Por esta razón, no se puede afirmar con rotundidad que un sistema sea el mejor de ambos.

Por otro lado, si se diferencian los resultados obtenidos en los ficheros de selección por el idioma y el género del periodista principal, también se observa que los valores de DER teniendo en cuenta todos los locutores son similares en ambos sistemas, como muestra la Figura 39. En cuanto al DER teniendo en cuenta únicamente el periodista, *config_elbow* vuelve a ofrecer mejores resultados que el resto de configuraciones, como se puede apreciar en la Figura 40, principalmente en la categoría de periodista principal hombre – euskera. Como se ha comentado anteriormente, existe una alta desviación estándar en los resultados de DER obtenidos, por lo que tampoco se puede asegurar que un sistema sea el mejor para determinado género o idioma del periodista.

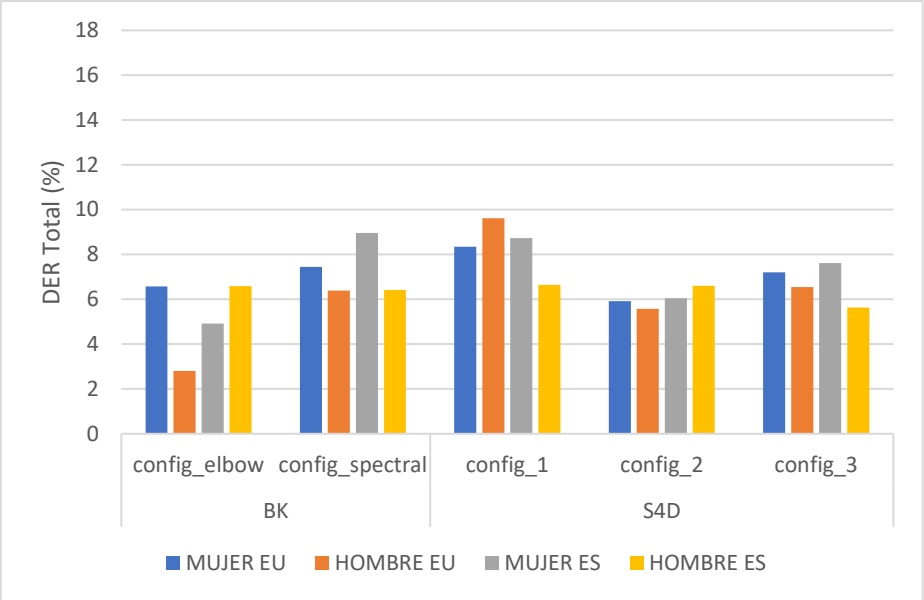


Figura 39: DER promedio total categorizado por idioma/género

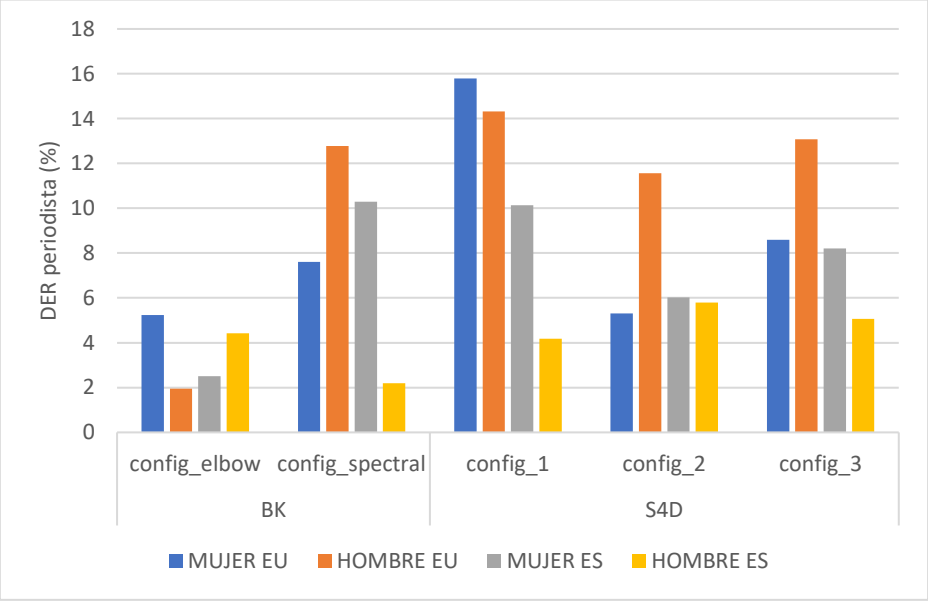


Figura 40:DER promedio periodista categorizado por idioma/género

Otro aspecto a tener en cuenta en la elección de los sistemas es el tiempo de ejecución. En la configuración con que menor DER se ha obtenido en cada sistema (*config_elbow* utilizando *binary keys* y *config_2 en s4d*), los tiempos de ejecución han sido de 104 segundos en *config_elbow* y de 87 segundos en *config_2*, en un total de 73 ficheros. Por tanto, en este caso emplear *config_elbow* ha supuesto un aumento de tiempo de ejecución del 19.5% respecto a *config_2* de *s4d*. El tiempo de ejecución es un aspecto a considerar al realizar la diarización sobre una gran cantidad de ficheros.

10. Planificación

En este apartado se describen las fases seguidas para la realización del Trabajo de Fin de Máster (TFM). Se detallan los Paquetes de Trabajo (P.T.), las tareas que los forman (T), la duración de las tareas y los recursos técnicos y humanos empleados.

10.1 Equipo de Trabajo

El equipo de trabajo está formado por una ingeniera senior, que supervisa el desarrollo del proyecto y de un ingeniero junior, que se encarga de la realización del proyecto y de la elaboración de la documentación final. Las personas que han formado parte del proyecto son las siguientes:

Ingeniera Senior: Eva Navas

Ingeniero Junior: Aingeru Buruchaga

10.2 Definición de paquetes de trabajo

P.T.1: Gestión del proyecto

T.1.1. Seguimiento del proyecto

Duración: 26 semanas

Descripción: Se realizan reuniones periódicas para la evaluación de las tareas y supervisar que se han completado satisfactoriamente.

Recursos humanos: Ingeniera Senior e Ingeniero Junior.

Recursos técnicos: Ordenador

T.1.2. Determinación del plan de trabajo

Duración: 1 semana

Descripción: Establecer el plan de trabajo a seguir para cumplir los objetivos del proyecto.

Recursos humanos: Ingeniera Senior e Ingeniero Junior.

Recursos técnicos: Ordenador

P.T.2: Estudio previo

T.2.1. Estudio del Estado del Arte

Duración: 4 semanas

Descripción: Estudio de las características, etapas y objetivos de la diarización de locutores y estudio del procedimiento para la evaluación de los resultados.

Recursos humanos: Ingeniero Junior.

Recursos técnicos: Ordenador

T.2.2. Estudio de la base de datos del proyecto

Duración: 1 semana

Descripción: Estudio de la base de datos utilizada en el proyecto.

Recursos humanos: Ingeniero Junior.

Recursos técnicos: Ordenador

T.2.3. Estudio y selección de sistemas de diarización

Duración: 2 semanas

Descripción: Estudio de los sistemas de diarización existentes y su adecuación a la base de datos del proyecto. Se seleccionan los sistemas a utilizar en el proyecto.

Recursos humanos: Ingeniero Junior.

Recursos técnicos: Ordenador

P.T.3: Implementación de los sistemas de diarización

T.3.1. Implementación de los sistemas con la configuración original

Duración: 3 semanas

Descripción: Implementación de los sistemas de diarización con los parámetros por defecto.

Recursos humanos: Ingeniero Junior

Recursos técnicos: Ordenador

T.3.2. Adaptación de los sistemas a la base de datos del proyecto

Duración: 3 semanas

Descripción: Configuración de diferentes parámetros y modificaciones sobre los sistemas de diarización originales con el objetivo de determinar la configuración óptima para la base de datos utilizada.

Recursos humanos: Ingeniero Junior

Recursos técnicos: Ordenador

P.T.4: Evaluación de los sistemas optimizados

T.4.1. Selección de ficheros y obtención de ficheros RTTM de referencia

Duración: 1 semanas

Descripción: Selección de los ficheros de audio sobre los que se realizan las evaluaciones y obtención de los ficheros RTTM de referencia para determinar el error cometido.

Recursos humanos: Ingeniero Junior

Recursos técnicos: Ordenador

T.4.2. Análisis del rendimiento de los sistemas

Duración: 3 semanas

Descripción: Análisis de los resultados obtenidos con las diferentes configuraciones, con el objetivo de determinar el sistema y la configuración más adecuados en función de los resultados de diarización y del tiempo requerido por cada sistema.

Recursos humanos: Ingeniero Junior

Recursos técnicos: Ordenador

P.T.5: Documentación

T.5.1. Redacción del informe final

Duración: 11 semanas

Descripción: Documentar el procedimiento seguido durante el proyecto, así como las configuraciones y pruebas realizadas y los resultados obtenidos.

Recursos humanos: Ingeniero Junior.

Recursos técnicos: Ordenador

10.3 Hitos

Para controlar que las diferentes etapas se cumplen en los plazos establecidos, se establecen los siguientes hitos(H):

H.1. Definición del proyecto. Semana 1

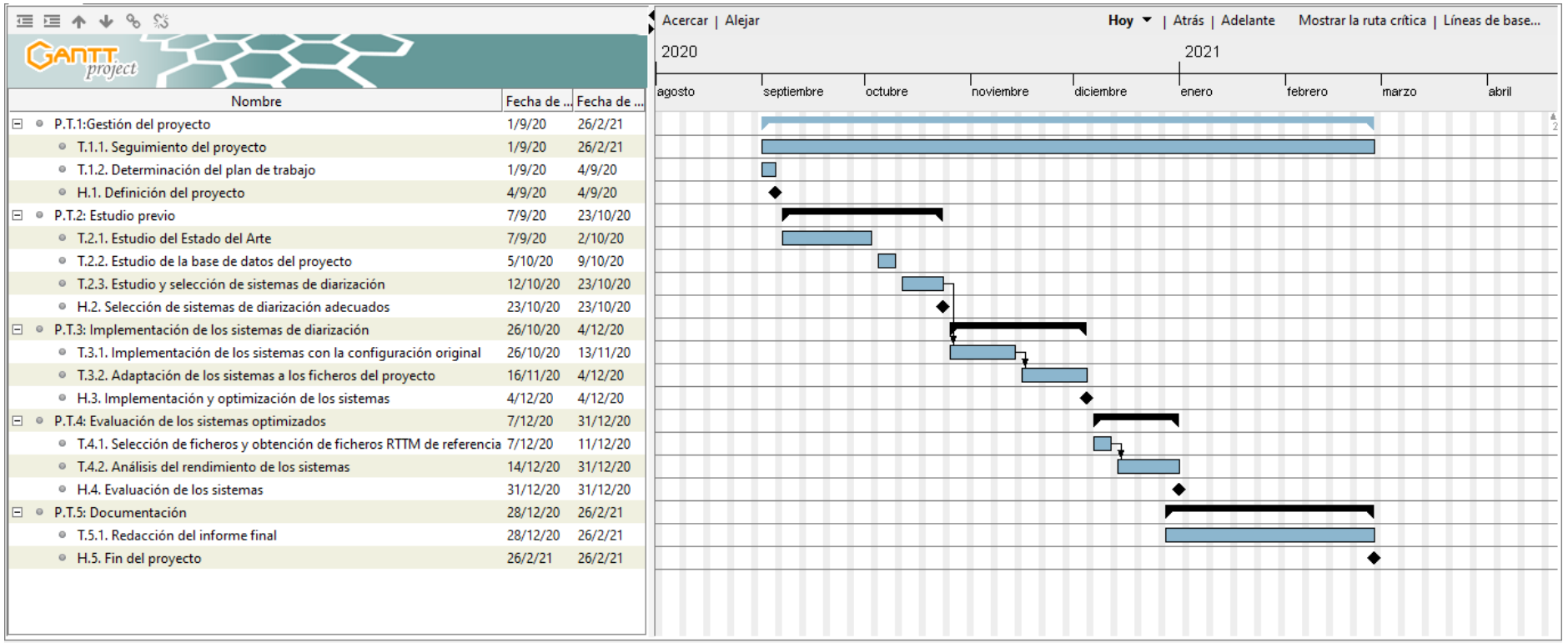
H.2. Selección de sistemas de diarización adecuados para la base de datos. Semana 8

H.3. Implementación y optimización de los sistemas. Semana 14.

H.4. Evaluación de los sistemas. Semana 18

H.5. Fin del proyecto. Semana 26

10.4 Diagrama de Gantt



Nombre	Fecha de ...	Fecha de ...
P.T.1:Gestión del proyecto	1/9/20	26/2/21
T.1.1. Seguimiento del proyecto	1/9/20	26/2/21
T.1.2. Determinación del plan de trabajo	1/9/20	4/9/20
H.1. Definición del proyecto	4/9/20	4/9/20
P.T.2: Estudio previo	7/9/20	23/10/20
T.2.1. Estudio del Estado del Arte	7/9/20	2/10/20
T.2.2. Estudio de la base de datos del proyecto	5/10/20	9/10/20
T.2.3. Estudio y selección de sistemas de diarización	12/10/20	23/10/20
H.2. Selección de sistemas de diarización adecuados	23/10/20	23/10/20
P.T.3: Implementación de los sistemas de diarización	26/10/20	4/12/20
T.3.1. Implementación de los sistemas con la configuración original	26/10/20	13/11/20
T.3.2. Adaptación de los sistemas a los ficheros del proyecto	16/11/20	4/12/20
H.3. Implementación y optimización de los sistemas	4/12/20	4/12/20
P.T.4: Evaluación de los sistemas optimizados	7/12/20	31/12/20
T.4.1. Selección de ficheros y obtención de ficheros RTTM de referencia	7/12/20	11/12/20
T.4.2. Análisis del rendimiento de los sistemas	14/12/20	31/12/20
H.4. Evaluación de los sistemas	31/12/20	31/12/20
P.T.5: Documentación	28/12/20	26/2/21
T.5.1. Redacción del informe final	28/12/20	26/2/21
H.5. Fin del proyecto	26/2/21	26/2/21

Figura 41:Diagrama de Gantt

11. Aspectos económicos

A continuación, se detalla el coste total del proyecto, dividiendo los costes en: horas internas, amortizaciones y gastos.

11.1 Horas internas

Se detallan los costes relacionados con el personal que ha participado en el proyecto.

Nombre	Puesto	Coste (€/h)	Horas	Total (€)
Eva Navas	Ingeniera Senior	50	50	2500
Aingeru Buruchaga	Ingeniero Junior	20	600	12000
TOTAL				**14500**

Tabla 22: Coste Horas internas

11.2 Amortizaciones

Se detallan los recursos necesarios a lo largo del proyecto. Se tiene en cuenta el coste correspondiente al tiempo de uso en este proyecto.

Concepto	Coste Inicial (€)	Vida útil	Tiempo de uso	Total (€)
Ordenador Ing. Senior	1000	5 años	6 meses	100
Ordenador Ing. Junior	700	5 años	6 meses	70
Licencia de Office	100	5 años	6 meses	10
TOTAL				**180**

Tabla 23: Coste Amortizaciones

11.3 Gastos

Se detallan los gastos en material utilizado exclusivamente para la realización de este proyecto.

Concepto	Coste (€)
Material de oficina	30
Electricidad	20
Conexión a Internet	100
TOTAL	**150**

Tabla 24: Coste Gastos

11.4Resumen

El coste total del proyecto es la suma de los costes calculados en los apartados anteriores.

Concepto	Coste (€)
Horas Internas	14500
Amortizaciones	180
Gastos	150
TOTAL	**14830**

Tabla 25: Resumen del presupuesto final

12. Conclusiones

Mediante este Trabajo de Fin de Máster se han analizado dos sistemas de diarización de locutores y se ha evaluado su rendimiento en ficheros de audio de la base de datos utilizada. En primer lugar, se ha estudiado la base de datos y se ha comprobado su adecuación para emplearla en el desarrollo de sistemas TTS, debido a la calidad de las voces de los periodistas y la gran cantidad de ficheros disponibles tanto en euskera como en castellano.

Por otro lado, tras el estudio de diferentes sistemas de diarización, se han seleccionado dos sistemas para realizar el proceso de diarización de locutores con los ficheros de la base de datos. Además, se ha conseguido modificarlos para obtener mejores resultados respecto a los sistemas originales.

Se han evaluado diferentes configuraciones de los sistemas y se ha hecho una comparativa entre sistemas para determinar con qué configuración se han obtenido mejores resultados. Esta última etapa de evaluación ha demostrado que ambos sistemas son adecuados para la diarización de locutores, obteniéndose valores de error bajos en el proceso de diarización.

Por todo ello, ambos sistemas de diarización pueden ser utilizados en ficheros de audio que contengan voces de diferentes locutores y conseguir audios con la voz de un único locutor para desarrollar un sistema TTS.

Un último punto a destacar es el potencial que pueden llegar a tener los dos sistemas de diarización utilizados. Ambos sistemas han ido evolucionando durante los últimos años, introduciendo nuevos métodos y mejoras. Por tanto, la introducción de nuevas técnicas en las diferentes etapas de la diarización, unido a los *challenges* organizados para poder probar las nuevas mejoras implementadas, hará que en un futuro el rendimiento los sistemas sea todavía mejor.

13. Referencias

[1] Prahallad, K. Automatic building of synthetic voices from audio books. Tesis Doctoral. Nagoya Institute of Technology. Julio, 2010.

[2] Kuligowska, K., Kisielewicz, P. Wlodarz, A. Speech synthesis systems: disadvantages and limitations. International Journal of Engineering and Technology. 7 (pp. 234 - 239). 2018.

[3] Tavarez, D. Técnicas de mejora del rendimiento de los sistemas de diarización de locutores. Tesis doctoral. Universidad del País Vasco (UPV/EHU). Diciembre, 2016.

[4] Delgado, H. Fast cross-session speaker diarization. Tesis Doctoral. Universidad Autónoma de Barcelona. Junio, 2015.

[5] Universidad de Las Palmas de Gran Canaria. Características estadísticas de la señal de voz. Página web. https://www2.ulpgc.es/hege/almacen/download/23/23210/teoriavocoderlpc.pdf Accedido Febrero, 2021.

[6] Bimbot, F., Bonastre, J.F., Fredouille C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-García, J., Petrovska – Delacrétaz, D., Reynols, D.A., A Tutorial on Text-Independent Speaker Verification. "EURASIP journal on Advances in Signal Processing 2004:4. (pp. 430 – 451). Abril, 2004.

[7] Hanilçi, C., Ertas, F. Optimizing acoustic features for source cell-phone recognition using speech signals. Proceedings of the First ACM Workshop on Information Hiding and Multimedia Security - IH&MMSec '13, (pp. 141-148). Junio, 2013.

[8] Zhou, X., Garcia-Romero, D., Duraiswami, R., Espy-Wilson, C., Shamma, S. Linear versus mel frequency cepstral coefficients for speaker recognition. 2011 IEEE Workshop on Automatic Speech Recognition & Understanding, (pp. 559-564). Diciembre, 2011.

[9] Hönig, F., Stemmer G., Hacker, C. & Brugnara, F. Revising Perceptual Linear Prediction (PLP). (pp. 2997-3000). 2005

[10] Kumar S. Understanding K-Means, K-Means++ and K-Medoids Clustering Algorithms. Towardsdatascience. Página Web. https://towardsdatascience.com/understanding-k-means-k-means-and-k-medoids-clustering-algorithms-ad9c9fbf47ca Junio, 2011.

[11] Fiscus, J. G., Radde N., Garofolo J.S., Le A., Ajot J., Laprun C. "The rich transcription 2005 spring meeting recognition evaluation." In International Workshop on Machine Learning for Multimodal Interaction, (pp. 369-389). Springer. 2005.

[12] Fiscus, J. G., Ajot J., Martial M., Garofolo J.S. "The rich transcription 2006 spring meeting recognition evaluation." In International Workshop on Machine Learning for Multimodal Interaction, (pp. 309-322). Springer. 2006.

[13] Fiscus, J. G., Ajot J., Garofolo J.S. "The rich transcription 2007 meeting recognition evaluation." In Multimodal Technologies for Perception of Humans, (pp. 373-389). Springer. 2007.

[14] Murugan, B. Audio Processing and Remove Silence using Python. Medium. 2020.

[15] Dempster, A. P., Laird, N. M., & Rubin, D. B. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society: Series B (Methodological), 39(1), 1-22. 1997.

[16] Reynolds, D. A. Comparison of background normalization methods for text-independent speaker verification. Fifth European Conference on Speech Communication and Technology. 1997.

[17] Anguera X., Bonastre J.-F., A novel speaker binary key derived from anchor models. *Interspeech*. 2010

[18] Nguyen, T. H., Chng, E. S., Li, H. T-test distance and clustering criterion for speaker diarization. In Ninth Annual Conference of the International Speech Communication Association. 2008.

[19] Delgado, H., Anguera, X., Fredouille, C., Serrano, J. Fast single-and cross-show speaker diarization using binary key speaker modeling. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 23(12), (pp. 2286-2297). 2015

[20] Ryant, N., Church, K., Cieri, C., Cristia, A., Du, J., Ganapathy, S., Liberman, M. First DIHARD challenge evaluation plan. 2018, tech. Rep. 2018.

[21] Patino, J., Delgado, H., & Evans, N. W. The EURECOM Submission to the First DIHARD Challenge. INTERSPEECH (pp. 2813-2817). Septiembre, 2018.

[22] Brown, J. C. Calculation of a constant Q spectral transform. The Journal of the Acoustical Society of America, 89(1), (pp. 425-434). 2018.

[23] Wang, Q., Downey, C., Wan, L., Mansfield, P. A., & Moreno, I. L. Speaker diarization with LSTM. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 5239-5243). Abril, 2018.

[24] Broux, P. A., Desnous, F., Larcher, A., Petitrenaud, S., Carrive, J., Meignier, S. S4D : Speaker Diarization Toolkit in Python. Interspeech. Septiembre, 2018.

[25] Stafylakis, T., Katsouros, V., Carayannis, G. The segmental Bayesian Information Criterion and its applications to Speaker diarization. IEEE Journal of Selected Topics in Signal Processing, 4(5), (pp. 857-866). 2010

Anexo I. Selección de ficheros para la evaluación de los sistemas

Por medio de este anexo se recogen las características principales de los ficheros de audio utilizados en la evaluación de los sistemas de diarización de locutores. Entre la información mostrada para cada fichero, se muestra el identificador del fichero, el género e idioma del periodista, la duración del fichero original, la duración tras el procesado para eliminar silencios, el número de locutores existentes, el orden seguido por los locutores y la identificación del género de los locutores que no son el periodista.

En los ficheros de audio en los que el periodista habla en castellano, el resto de locutores habla en castellano. En el caso de los ficheros en los que el periodista habla en euskera, los locutores hablan tanto en euskera como en castellano, por lo que en su descripción se ha especificado el idioma utilizado por los locutores.

Abreviaturas: P: Periodista, L1: Locutor 1, LN: Locutor N

Audio ID	Género	Idioma	Duración Original	Duración Sin Silencios	Num Speakers	Secuencia	Descripción
Audio_001	Mujer	Castellano	1:29	1:14	5	P - L1 - L2 - P - L3 - P - L4 - P	1º Periodista 2º Mujer 3º Hombre 4º Periodista 5º Mujer 6º Periodista
Audio_002	Mujer	Castellano	1:11	0:59	2	P - L1 –P	1ºPeriodista 2º Hombre 3º Periodista
Audio_003	Mujer	Castellano	1:10	0:58	2	P - L1 –P	1º Periodista 2ºHombre 3º Periodista
Audio_004	Mujer	Castellano	1:24	1:08	2	P - L1 - P -L1 -P	1º Periodista introduce noticia. 2ºHombre de mediana edad 3º Periodista habla 4º Hombre de mediana edad 5º Periodista finaliza noticia
Audio_005	Mujer	Castellano	1:25	1:12	2	P - L1 - P - L1- L1 -P	1º Periodista 2º Hombre 3º Periodista 4º Hombre 5º Periodista 6º Hombre 7º Periodista
Audio_006	Mujer	Castellano	1:40	1:16	4	P - L1 - P - L2 - P - L3 - L2 - P	1º Periodista 2º Hombre 3º Periodista 4º Hombre 5º Periodista 6º Hombre 7º Hombre 8º Periodista
Audio_007	Mujer	Castellano	1:33	1:19	3	P - L1 - P - L2 - P L1 -P	1º Periodista 2º Hombre 3º Periodista 4º Mujer 5º Periodista 6º Hombre 7º Periodista
Audio_008	Mujer	Castellano	1:32	1:23	5	P - L1 -P - L2 - P - L3 - P - L4 – P	1º Periodista 2º Hombre 3º Periodista 4º Hombre 5º Periodista 6º Mujer 7º Periodista 8º Hombre 9º Periodista
Audio_009	Mujer	Castellano	1:25	1:11	4	P - L1 - L2 - P - L3 - P	1º Periodista 2º Mujer 3º Mujer (catalán) 4º Periodista 5º Mujer 6º Periodista
Audio_010	Mujer	Castellano	1:08	1:00	2	L1 - P - L1 -P - L1	1º Hombre de mediana edad habla. 2º Periodista habla. 3º Hombre de mediana edad habla 4º Periodista habla 5º Hombre de mediana edad habla
Audio_011	Mujer	Castellano	0:49	0:43	2	P - L1 –P	1º Periodista introduce noticia. 2º Mujer de mediana edad habla. 3º Periodista finaliza noticia
Audio_012	Mujer	Castellano	1:52	1:38	12	P - L1 -L2 - L3 - L4 - P - L5 - L6 - P -L7 -L8 -L9 - L10 - P -L11 –P	1º Periodista introduce noticia. 2º,3º,4º,5º - Diferentes personas (3 chicas jóvenes y 1 chico joven) 6º Periodista habla 7ºMujer de mediana edad 8º Hombre de mediana edad 9º Periodista 10º,11º,12,13º Diferentes personas (3 chicas jóvenes y 1 chico joven) 14º Periodista 15º Mujer mediana edad 16º Periodista
Audio_013	Mujer	Castellano	1:32	1:19	4	P - L1 - P - L2 -P - L3	1º Periodista introduce noticia. 2º Hombre de mediana edad habla. 3º Periodista habla. 4º Hombre de mediana edad habla. 5º Periodista habla. 6º Mujer habla
Audio_014	Mujer	Castellano	0:51	0:28	4	P -L1 -L2 -L3	1º Periodista introduce noticia 2º Mujer de mediana edad 3º Mujer joven 4º Hombre joven
Audio_015	Mujer	Castellano	1:16	0:59	2	P - L1 -P -L1 -P -L1 -P	1º Periodista introduce noticia 2º Mujer joven 3º Periodista 4º Mujer joven 5º Periodista 6º Mujer joven 7º Periodista

Audio_016	Mujer	Castellano	1:12	1:02	6	P - L1 -P -L1 - P -L2 - P -L3 -L4 -L5 –P	1º Periodista introduce noticia. 2º Hombre de mediana edad. 3º Periodista. 4º Hombre de edad avanzada habla 5º Periodista. 6º Hombre de edad avanzada. 7º Periodista 8º Hombre joven 9º Mujer de mediana edad 10º Mujer de mediana edad 11º Periodista finaliza noticia
Audio_017	Mujer	Castellano	1:16	1:04	2	P - L1 – P	1º Periodista introduce noticia 2º Mujer mediana edad 3º Periodista
Audio_018	Mujer	Castellano	1:02	0:50	7	P - L1 - L2 -L3 - L4 - L5 -L6 -P	1º Periodista introduce noticia. 2º Hombre de avanzada edad 3º Mujer de mediana edad 4º Mujer de mediana edad 5º Hombre de mediana edad 6º Mujer de mediana edad 7º Hombre de mediana edad 8º Periodista
Audio_019	Mujer	Castellano	1:22	1:05	3	P - L1 - P -L2 - P - L2	1º Periodista introduce noticia. 2º Hombre habla 3º Periodista 4º Niño habla en euskera 5º Periodista 6º Niño habla en euskera
Audio_101	Hombre	Castellano	1:23	1:04	3	P - L1 - P - L2 -P	1ºEl periodista introduce el contexto de la noticia. 2ºUn hombre joven habla. 3º habla el periodista. 4º Hombre joven habla sin ruido de fondo. 5º Periodista finaliza la noticia
Audio_102	Hombre	Castellano	1:02	0:50	2	P - L1 – P	1º Periodista 2º Hombre 3º Periodista
Audio_103	Hombre	Castellano	1:25	0:59	3	P - L1 - L2 - P	1º Periodista 2º Hombre 3ºHombre 4º Periodista
Audio_104	Hombre	Castellano	1:20	1:12	5	L1 - L2 - L3 -P - P - L4	1º Mujer 2ºHombre (fragmento corto)3ºMujer 4º Periodista 5º Periodista (Diferente calidad de audio)6ºMujer
Audio_105	Hombre	Castellano	1:19	0:55	3	P - L1 - P - L2 -P	1º Periodista 2º Hombre 3º Periodista 4º Hombre 5º Periodista
Audio_106	Hombre	Castellano	1:14	1:01	5	P - L1 - P - L2 - L3 - P - L4 - P	1º Periodista 2º Hombre 3º Periodista 4ºNiño 5º Hombre joven 6º Periodista 7º Hombre 8º Periodista
Audio_107	Hombre	Castellano	1:13	0:51	2	P - L1 – P	1º Periodista 2ºHombre 3º Periodista
Audio_108	Hombre	Castellano	1:38	1:20	3	P - L1 - P - L1 - P - L1 -P - L1 - P - L2 – P	1º Periodista 2º Hombre 3º Periodista 4º Hombre 5º Periodista 6º Hombre 7º Periodista 8º Hombre 9º Periodista 10º Mujer 11º Periodista
Audio_109	Hombre	Castellano	1:20	1:15	2	P - L1 - P - L1 - P -L1 - P	1º Periodista 2º Hombre 3º Periodista 4º Hombre 5º Periodista 6º Hombre 7º Periodista
Audio_110	Hombre	Castellano	1:27	1:15	4	L1 - P -P - L1 - P -L2 - L3 - P	1º Hombre 2º Periodista 3º Periodista 4º Hombre 5ºPeriodista 6º Hombre 7º Mujer 8º Periodista
Audio_111	Hombre	Castellano	1:04	0:54	3	P - L1 - P -L2 - P - L2	1º Periodista introduce noticia 2º Hombre en inglés (fragmento muy corto) 3º Periodista 4º Hombre en castellano 5º Periodista finaliza noticia
Audio_112	Hombre	Castellano	1:37	1:14	3	L1 - P - L1 - P -L2 - P -L2	1º Hombre 2º Periodista 3º Hombre 4º Periodista 5º Hombre (fragmento corto) 6º Periodista 7º Hombre
Audio_113	Hombre	Castellano	1:28	1:11	3	P - L1 - P - L1 - P - L2 - P	1º Periodista introduce noticia 2º Hombre 3º Periodista 4º Hombre 5º Periodista 6º Hombre 7º Periodista

Audio_114	Hombre	Castellano	1:23	1:11	2	P - L1 - P - L1 -P - L1 -P - L1	1º Periodista introduce la noticia. 2º Hombre joven habla. 3º Periodista habla. 4º Hombre joven habla. 5º Periodista habla. 6º Hombre joven habla. 7º Periodista habla. 8º Hombre joven habla
Audio_115	Hombre	Castellano	1:15	1:03	2	P - L1 – P	1º Periodista comienza describiendo la noticia. 2º Hombre habla en portugués con ligero ruido de fondo. 3º Habla el periodista.
Audio_116	Hombre	Castellano	1:45	1:19	3	P - L1 - P -L2 -P	1º Periodista introduce noticia. 2º Habla hombre de mediana edad 3º Habla periodista. 4º Habla hombre de mediana edad. 5º Periodista finaliza noticia
Audio_117	Hombre	Castellano	1:34	1:22	3	P - L1 - P -L2 -P	1º Periodista introduce la noticia. 2º Hombre joven habla sobre el tema. 3º Habla periodista. 4º Hombre de mediana edad. 5º Periodista finaliza la noticia
Audio_201	Mujer	Euskera	1:13	1:00	2	P - L1 – P	1º Periodista introduce noticia 2º Mujer (EUSK) 3º Periodista
Audio_202	Mujer	Euskera	1:24	1:14	3	P -L1 - P - P - L2 - P	1º Periodista introduce noticia 2º Hombre (EUSK) 3ºPeriodista 4º Periodista 5º Mujer (EUSK) 6º Periodista
Audio_203	Mujer	Euskera	1:18	0:57	3	P - L1 - P - L1 - P - L2 - P	1º Periodista 2º Hombre (EUSK) 3º Periodista 4º Hombre (EUSK) 5ºPeriodista 6º Hombre 7º Periodista
Audio_204	Mujer	Euskera	1:11	0:59	10	L1 -L2 - L1 - L2 - L1 - P - L3 - L4 - L5 - L6 - P - L7 - P - L8 -L9 –P	1º Niño pequeño (EUSK) 2º Mujer (EUSK), fragmento corto 3º Niño pequeño (EUSK), fragmento corto 4º Mujer (EUSK), fragmento corto 5º Niño pequeño (EUSK), fragmento corto 6º Periodista 7º Niño, fragmento corto 8º Niño, fragmento corto 9º Hombre 10º Mujer 11º Periodista 12º Niño 13º Hombre 14º Niño pequeño, 15º Niña pequeña 16º Niña pequeña 17º Periodista
Audio_205	Mujer	Euskera	1:16	0:58	2	P - L1 - P -L1	1º Periodista 2º Niña pequeña (EUSK) 3º Periodista 4º Niña pequeña (EUSK)
Audio_206	Mujer	Euskera	1:00	0:50	2	P - L1 - P -L1 - P	1º Periodista 2º Hombre (EUSK) 3º Periodista 4º Hombre (EUSK) 5º Periodista
Audio_207	Mujer	Euskera	1:15	1:06	8	P - L1 - P - L2 -L3 -L4 - L6 - P - L7 – P	1º Periodista 2º Mujer (EUSK) 3º Periodista 4º Mujer (EUSK) 5ºHombre (CAST) 6º Mujer (CAST) 7º Periodista 8º Mujer (EUSK) 9º Periodista
Audio_208	Mujer	Euskera	1:20	0:45	2	P - L1 – P	1º Periodista 2º Hombre (EUSK) 3º Periodista
Audio_209	Mujer	Euskera	1:37	1:25	3	P -L1 - P - L1 -P - L2 - P	1º Periodista 2º Hombre (EUSK) 3º Periodista 4º Hombre (EUSK) 5º Periodista 6ºMujer (CAST), fragmento corto 7º Periodista (con voz de fondo, traduciendo)
Audio_210	Mujer	Euskera	1:21	1:09	5	P -L1 -L2 P -L3 -L4 -P	1º Periodista introduce noticia. 2º Mujer habla (CAS) 3ºHombre habla (EUSK) 4º Periodista habla 5º Mujer habla (CAS) 6º Hombre habla (EUSK) 7º Periodista

Audio_211	Mujer	Euskera	1:09	1:01	2	P - L1 - P -L1 - P - L1 -P	1º Periodista 2º Hombre (EUSK) 3º Periodista 4º Hombre (EUSK) 5º Periodista 6º Hombre (EUSK) 7º Periodista
Audio_212	Mujer	Euskera	1:06	0:47	2	P - L1 - P -L1 - P	1º Periodista 2º Mujer (CAST) 3º Periodista 4º Mujer (CAST) 5º Periodista
Audio_213	Mujer	Euskera	1:04	0:55	2	P - L1 – P	1º Periodista introduce noticia 2º Mujer (EUSK) 3ºPeriodista
Audio_214	Mujer	Euskera	1:25	1:08	2	P - L1 - P -L1 - P	1º Periodista 2º Hombre (EUSK) 3º Periodista 4º Hombre (EUSK) 5º Periodista
Audio_215	Mujer	Euskera	1:18	1:05	2	P -L1 - P - L1 -P	1º Periodista introduce noticia 2º Hombre de mediana edad (CAS) 3º Periodista 4º Hombre de mediana edad (CAS) 5º Periodista finaliza noticia
Audio_216	Mujer	Euskera	1:53	1:22	4	P -L1 -P -L1 -P -L1 - P -L2 -P - L3 –P	1º Periodista introduce noticia 2º Mujer (EUSK) 3º Periodista 4º Mujer (EUSK) 5º Periodista 6º Mujer (EUSK) 7º Periodista 8º Mujer (CAS) 9º Periodista 9º Mujer (EUSK) 10º Periodista
Audio_217	Mujer	Euskera	1:00	0:46	2	P -L1 -P -1 -P	1º Periodista introduce noticia 2º Mujer (EUSK)- tramos corto 3º Periodista 4º Mujer (EUSK) 5º Periodista
Audio_218	Mujer	Euskera	1:13	1:04	3	P - L1 -P -L2 - P -P	1º Periodista introduce noticia 2º Mujer (CAS) 3º Periodista 4º Hombre (CAS) 5º Periodista 6º Periodista - con ruido de ambiente exterior
Audio_219	Mujer	Euskera	1:18	0:57	9	P - L1 - L2 - L3 - P - L4 -L5 - P - L6 - L7 - L8 -P	1º Periodista 2º Mujer (CAST) 3º Mujer(CAST) 4º Hombre (CAST) 5º Mujer(CAST) 6º Periodista 7º Mujer (CAST) 8º Mujer (CAST) 9º Hombre (CAST) 10º Periodista
Audio_301	Hombre	Euskera	1:45	1:33	8	P - L1 - L2 -L3 - P -L4 - P - L5 - P - L6 -P -L7 -P	1º Periodista 2º Hombre(EUSK) 3º Mujer(EUSK)4ºMujer (EUSK) 5º Periodista 6º Mujer(EUSK) 7º Periodista 8º Mujer(EUSK) 9º Periodista 10º Hombre(EUSK)11º Periodista
Audio_302	Hombre	Euskera	1:32	1:08	2	P - L1 - P -L1 -P	1º Periodista introduce noticia 2º Hombre (EUSK) 3º Periodista 4º Hombre (EUSK) 5º Periodista
Audio_303	Hombre	Euskera	1:08	0:56	3	P - L1 -P - L2 -P	1º Periodista introduce noticia 2º Mujer (CAST) 3º Periodista 4º Hombre (CAST) 5º Periodista
Audio_304	Hombre	Euskera	0:59	0:44	2	P - L1 – P	1º Periodista 2ºHombre(EUSK) 3ºPeriodista
Audio_305	Hombre	Euskera	1:09	0:52	3	P - L1 - P - L2 - P	1º Periodista introduce noticia 2º Hombre(EUSK) 3º Periodista 4º Hombre (EUSK) 5º Periodista
Audio_306	Hombre	Euskera	0:57	0:51	2	P - L1 - P -L1 - P	1º Periodista 2º Mujer(CAST) 3º Periodista 4º Mujer(CAST) 5º Periodista
Audio_307	Hombre	Euskera	1:28	1:06	3	P -L1 - P -L2 -P	1º Periodista introduce noticia 2º Hombre (EUSK) 3º Periodista 4º Mujer (EUSK) 5º Periodista
Audio_308	Hombre	Euskera	1:13	0:55	3	L1 - P - L1 -P - L2 -P	1º Mujer (EUSK) 2º Periodista 3ºMujer (EUSK) 4º Periodista 5º Hombre (EUSK) 6º Periodista

Audio_309	Hombre	Euskera	1:18	1:10	4	P - L1 - P - L2 - P - L3 - P -P	1º Periodista 2º Mujer (EUSK)(hay una parte con ruido de fondo) 3º Periodista 4º Mujer (EUSK) 5º Hombre (CAST) 6º Periodista (pero como si fuera un corte) 7º Periodista
Audio_310	Hombre	Euskera	1:10	0:50	3	P - L1 - P -L2- P	1º Periodista 2º Hombre(EUSK) 3º Periodista 4º Hombre(EUSK) 5º Periodista
Audio_311	Hombre	Euskera	1:14	1:00	2	P - L1 - P - L1 - P - L1 -P	1º Periodista introduce noticia 2º Hombre (EUSK) 3º Periodista 4º Hombre (EUSK) 5º Periodista 6º Hombre (EUSK) 7º Periodista
Audio_312	Hombre	Euskera	1:01	0:45	2	P - L1 - P - L1 -P	1º Periodista introduce noticia 2º Hombre (EUSK) 3º Periodista 4º Hombre (EUSK)
Audio_313	Hombre	Euskera	1:30	1:13	6	P - L1 -P - L2 - P - L3 - L4 - P - L5	1º Periodista 2º Niño (EUSK). Fragmento corto 3º Periodista 4º Hombre(EUSK) 5º Periodista 6º Hombre (EUSK) 7º Hombre (EUSK) 8º Periodista 9º Hombre (EUSK)
Audio_314	Hombre	Euskera	1:24	1:09	3	P - L1 - P -L2- P	1º Periodista 2º Hombre(EUSK) 3º Periodista 4º Hombre(EUSK) 5ºPeriodista
Audio_315	Hombre	Euskera	1:27	1:02	3	P - L1 - P -L2- P	1º Periodista 2º Hombre(EUSK) 3º Periodista 4º Hombre(EUSK) 5º Periodista
Audio_316	Hombre	Euskera	1:21	0:58	2	P - L1 - P -L1 - P	1ºPeriodista 2º Hombre(EUSK) 3º Periodista 4º Hombre(EUSK) 5º Periodista
Audio_317	Hombre	Euskera	1:21	1:02	2	P - L1 – P	1ºPeriodista 2º Hombre(CAST) 3ºPeriodista
Audio_318	Hombre	Euskera	1:08	0:59	3	P - L1 - P -L1 - P -L2	1º Periodista 2º Mujer(CAST) 3º Periodista 4ºMujer(CAST)5ºPeriodista 6º Mujer(CAST)