6 BASE DE DATOS Y SOFTWARE

En este capítulo describiremos la base de datos con la que hemos elaborado las pruebas experimentales a lo largo de este proyecto, así como los protocolos seguidos para la realización de las diferentes pruebas. La base de datos utilizada en este proyecto constituye el conjunto de datos necesarios para el funcionamiento del sistema de reconocimiento de locutor. Estos datos son los requeridos para las fases de desarrollo, entrenamiento y evaluación. Los protocolos son pautas y medidas objetivas que permiten saber de qué forma hemos llegado a los resultados obtenidos. Finalmente, describiremos el software empleado en este proyecto.

6.1 Evaluaciones NIST

El organismo norteamericano NIST (*National Institute of Standards and Technology*) organiza evaluaciones internacionales competitivas de tecnología de reconocimiento de locutor e idioma. En este proyecto se ha trabajado con las evaluaciones SRE (*Speaker Recognition Evaluation*) [NIST SRE] correspondiente al año 2005 [NIST SRE 2005].

Las evaluaciones NIST tienen un carácter abierto, en ellas participan grupos de investigación de todo el mundo. Su intención es establecer condiciones competitivas que permitan analizar el rendimiento de los diferentes sistemas involucrados. Una de las principales finalidades de este tipo de evaluaciones es poder comparar los distintos sistemas, técnicas y configuraciones de cada uno de los integrantes.

El protocolo de evaluación define la medida de rendimiento (ya comentamos en la sección 2.3.4 que el NIST evalúa a través de la función de coste) junto con los datos sobre los que realizar la evaluación. Es el mismo procedimiento para todos los integrantes y está definido por los datos de entrenamiento y de test, así como datos complementarios como los utilizados para la compensación, normalización, etc.

En las evaluaciones NIST SRE la tarea fundamental consiste en la verificación de un individuo a partir de una evaluación de prueba. Con respecto a 2005, tanto para los datos de entrenamiento como de evaluación disponemos de diferentes condiciones en función de la cantidad y tipo de datos. En concreto, para los datos de entrenamiento disponemos de 5 condiciones distintas que pueden variar desde 10 segundos de habla hasta 8 conversaciones de 5 minutos (aproximadamente 2.5 minutos para cada locutor). Para los datos de test, tenemos cuatro condiciones diferentes, entre los que nos podemos encontrar conversaciones de 10 segundos hasta 1 conversación de 5 minutos (unos 2.5 minutos de hablas por locutor). Todas estas condiciones pueden proporcionarse incluyendo datos de entrenamiento/test en dos canales (*4-wire*), en el cual el habla de los locutores de la conversación se encuentra en ficheros separados, ó en canales sumados (*2-wire*) en el que las conversaciones se encuentran mezcladas en un único fichero de audio.

Puesto que disponemos de 5 condiciones de entrenamiento y 4 de test, en total tenemos 20 condiciones de juicio o trial (enfrentamiento de un fichero de training y test) que resumimos a continuación mediante una tabla correspondiente al plan de evaluación del NIST durante el año 2005:

		Test Segment Condition			
		10 sec 2-chan	**1 conv 2-chan**	**1 conv summed-chan**	**1 conv aux mic**
Training Condition	**10 seconds 2-channel**	optional	optional	optional	optional
	1 conversation 2-channel	optional	**required**	optional	optional
	3 conversation 2-channel	optional	optional	optional	optional
	8 conversation 2-channel	optional	optional	optional	optional
	3 conversation summed-channel	optional	optional	optional	optional

Tabla 1. *Condiciones de entrenamiento y test en la evaluación [NIST SRE 2005]*

Tal y como podemos observar en la figura anterior, de las 20 condiciones, el NIST solamente te obliga a participar en una, que se corresponde con la de 2.5 minutos para entrenamiento y test, respectivamente (1conv4w-1conv4w), mientras que las restantes 19 aparecen como opcionales. En este proyecto, salvo en un caso excepcional que se mencionará en su momento, hemos llevado a cabo los experimentos y simulaciones con los datos pertenecientes a la condición obligatoria, es decir, ficheros de características 1conv4w tanto para el entrenamiento como para el test. También añadir que el NIST te permite trabajar por separado para hombres y mujeres (hay un conjunto de juicios para hombres y otra para mujeres). En este proyecto se ha trabajado únicamente con la opción de locutores masculinos. Finalmente comentar que cualquier simulación de este proyecto será enfrentado a la misma base de datos, es decir, a lo juicios del NIST de 2005.

Con respecto a la organización de los datos, el NIST envía tres directorios o carpetas. Una de ellas es la carpeta *train* (de entrenamiento). Esta carpeta cuenta con todas las grabaciones de entrenamientos (las que se usan para generar el modelo de locutor). Además, cuenta con 5 tipos de ficheros, correspondientes a las 5 posibles condiciones de entrenamiento, en donde se le asigna un identificador a cada una de las

locuciones. El fichero correspondiente a la opción 1conv4w contiene 274 referencias (274 locutores serán entrenados). La estructura de dicho fichero aparece reflejada en la figura siguiente:

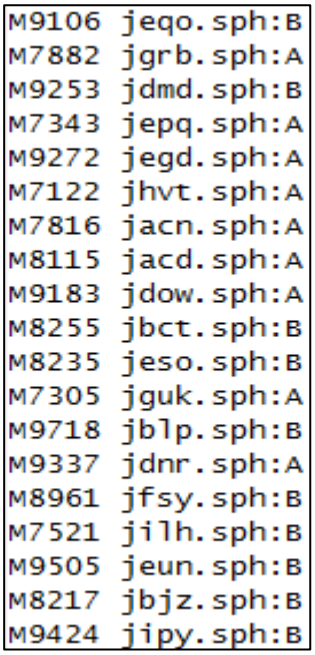

Figura 33. *Porción del fichero 1conv4w.trn correspondiente a la carpeta train de la evaluación del NIST de 2005.*

En dicha figura podemos apreciar como a cada locución se le asocia un identificador numérico acompañado de la letra M para indicar que se trata de locuciones pertenecientes a hombres (Male). Dicho identificador representará al modelo de locutor generado por cualquiera de las técnicas en la fase de entrenamiento.

La segunda carpeta recibe el nombre de *test*, y contiene todas las grabaciones de test que se enfrentarán a las locuciones de entrenamiento para verificar si son pronunciadas o no por la misma persona.

Finalmente, la tercera carpeta se denomina *trial* y está constituido por 20 ficheros correspondiente a las 20 combinaciones posibles entre tipos de segmentos de entrenamiento y de test. En ellos se enfrentan locuciones de entrenamiento y de evaluación. En la figura 34 se puede visualizar su estructura. Se ve como las locuciones de entrenamiento representadas mediante sus correspondientes identificadores numéricos son enfrentadas a locuciones almacenadas en la carpeta de *test*. De nuevo, la letra M entre medio es para enfatizar que se trata de un enfrentamiento entre hombres. El NIST también plantea juicios en donde en lugar de la letra M, nos podemos encontrar la letra F entre medio, indicando que la locución de test se encuentra almacenada en la sección correspondiente a mujeres. Esta opción no se ha barajado en este proyecto. Teniendo en cuenta que se ha empleado la base de datos *Det1 [NIST SRE 2005]*, la opción de locutores masculinos con la condición 1conv4w-1conv4w da lugar a 1231

juicios de tipo true-target (el segmento de entrenamiento y test pertenecen al mismo locutor) y 12317 juicios de tipo non-target, en el que los dos tipos de segmentos no son hablados por la misma persona.

```
M9170 M njsa A
M9170 M njoy B
M9170 M nanp B
M9170 M nkcl A
M9170 M nktv A
M9170 M nmwb B
M9170 M nhvh A
M9170 M ncah A
M9170 M nbsb B
M9170 M nibc A
M9170 M nkuq A
M9170 M nmnf B
M9170 M nbeh A
```

Figura 34. *Porción del fichero 1conv4w-1conv4w.ndx correspondiente a la carpeta trial de la evaluación del NIST de 2005*

Adicionalmente, en este proyecto se ha hecho uso de datos del NIST correspondiente a la evaluación de 2004 [NIST SRE 2004] como información adicional y complementaria para las técnicas que hemos implementado. Esta medida no supone ningún inconveniente en el sentido de que el NIST se compromete a que los individuos que participan un año en una evaluación no pueden participar al año siguiente, con lo cual tenemos la certeza absoluta de que los segmentos de audio que estamos incluyendo no pertenecen a locutores que participen en el año 2005. A continuación explicamos la información adicional que hemos tomado de la base de datos de 2004 y su correspondiente objetivo:

- 219 grabaciones de locutores que utilizamos para generar el UBM. No tiene ningún sentido utilizar los datos de la evaluación en la que participas para crear el UBM, puesto que el UBM es un modelo universal y puedes generarlo de manera independiente. Estas mismas grabaciones se utilizarán para representar a la cohorte de impostores necesaria en SVM.

- 116 segmentos de audio individuos (impostores) que nos permiten llevar a cabo el procedimiento de normalización de puntuaciones (Znorm, Tnorm, ZTnorm, Tznorm).

- Diferentes segmentos de voz (grabadas en diferentes condiciones de canal) para 124 locutores con el objetivo de resolver el problema de la variabilidad de intersección que tiene en cuenta tanto NAP como FA.

6.2 Software ALIZE

En este proyecto, todas las pruebas experimentales se han llevado a cabo mediante el software ALIZE [ALIZE], que se trata de una herramienta desarrollada y mantenida por LIA (*Laboratorie d'Informatique Avignon*) en la Universidad de Avignon. El paquete ALIZE ha sido implementado en un entorno de Linux y permite trabajar con sistemas de reconocimiento y autenticación biométrica. En concreto, dispone de herramientas y paquetes binarios para hacer todas las tareas necesarias en un sistema de reconocimiento de voz: generación de modelo, normalización, scoring, etc. En esta sección trataremos de mencionar y describir las funciones de ALIZE que hemos utilizado para implementar cada una de las técnicas.

En primer lugar, comentar que las grabaciones a través de conversaciones telefóncias enviadas por el NIST están codificadas en formato .sph (SPHERE file) [NIST File Format]. Atendiendo a todo el razonamiento teórico seguido a lo largo de este proyecto, lo primera operación necesaria dado un fichero de audio del NIST es su conversión a los parámetros MFCC. Para ello en este proyecto se hace uso de una herramienta adicional e independiente de ALIZE conocida como SPro [SPro Tools] que permite obtener los coeficientes cepstrales a través de la función *sfbcep*:

- *sfbcep* genera por cada trama de un fichero sph (10 ms) un vector de 60 componentes con el siguiente formato:

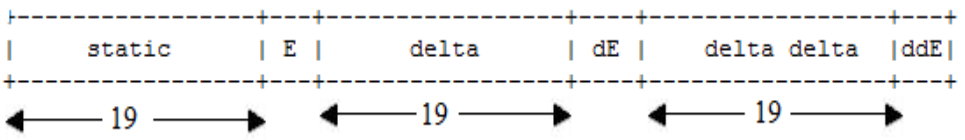

Figura 35. *Figura 35. Formato de los vectores cepstrales generados por SPro. Figura obtenida de [SPro Tools]*

En donde podemos ver que se generan 19 coeficientes estáticos (MFCC), 19 coeficientes de velocidad (MFCC-Delta) y otros 19 de aceleración (MFCC-Delta-Delta). Además, para cada uno de estos tres tipos de coeficientes, tenemos sus correspondientes energías, dando lugar por tanto a vectores cepstrales 60-dimensionales. Sin embargo, como veremos en el capítulo de los resultados, generalmente no se trabajan con los 60 componentes, y compararemos la eficiencia del sistema en función de la dimensión de los vectores cepstrales. Los coeficientes obtenidos con SPro son perfectamente compatibles con ALIZE. Para su correcto entendimiento, se ha mostrado a continuación el resultado de un fichero de salida correspondiente a la llamada a la función *sfbcep* en una de las simulaciones que se han llevado a cabo en este proyecto.

```
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tite.sph prms/04/tite_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tisj.sph prms/04/tisj_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tiqo.sph prms/04/tiqo_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tipw.sph prms/04/tipw_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tipa.sph prms/04/tipa_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tiok.sph prms/04/tiok_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/timr.sph prms/04/timr_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tikq.sph prms/04/tikq_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tigo.sph prms/04/tigo_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tift.sph prms/04/tift_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tife.sph prms/04/tife_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tieq.sph prms/04/tieq_LFCC.tmp.prm
./bin/sfbcep -F sphere -p 19 -e -D -A -k 0 -i 300 -u 3400 prms/sph/04/tieg.sph prms/04/tieg_LFCC.tmp.prm
```

Figura 36. *Fichero de salida tras calcular los coeficientes cepstrales a un conjunto de locuciones a través de la función sfbcep de SPro.*

En la figura anterior se observan algunas opciones interesantes de *sfbcep*. La opción –p 19 indica que queremos que nos calcule 19 coeficientes estáticos. –e indica que se debe de incluir la componente de energía (en concreto, el logaritmo de la energía). –D y –A son opciones que te permiten añadir los coeficientes de velocidad y aceleración, respectivamente. –k 0 establece el coeficiente de preénfasis a 0. Finalmente, -i 300 –u 3400 fijan el ancho de banda sobre el que se quiere calcular los coeficientes cepstrales. Puesto que las grabaciones del NIST provienen de conversaciones telefónicas, se trabaja en dicho ancho de banda. Los coeficientes cepstrales obtenidos de una locución .sph son almacenados en un fichero de formato .prm.

En cualquier prueba experimental correspondiente a este proyecto, tras calcular los coeficientes MFCC a través de la herramienta SPro (*sfbcep*), e independientemente de la técnica que se esté implementando, se hace una llamada a las dos siguientes funciones o paquetes implementados en ALIZE:

- *EnergyDetector*. Descarta las tramas ruidosas o silenciosas. Tal y como introdujimos en 3.3, permite trabajar en modo Mean SAD o Weight SAD. Sobre el valor de α_{SAD} se hablará en el apartado de simulaciones.

- *NormFeat*. Normaliza los parámetros mediante CMVN.

Además, independientemente de la técnica que implementemos, en cualquier experimento, también haremos uso de las siguientes funciones de ALIZE:

- *ComputeNorm*. Sirve para obtener la varianza y media de las puntuaciones obtenidas por los impostores en el procedimiento de normalización de puntuaciones.

- *ComputeTest.* Permite calcular la puntuación al enfrentar un fichero de entrenamiento y un fichero de salida.

- *Scoring*. El fichero de puntuaciones obtenido a través de la función anterior es transformado a un formato especificado por el NIST en el correspondiente plan de evaluación [NIST SRE 2005]. Como detalle puramente informativo, mostraremos a continuación un ejemplo del formato del fichero de resultado que teóricamente habría que enviar al NIST obtenido como resultado de la llamada a la función *Scoring* en una de las simulaciones que se han llevado a cabo en este proyecto:

```
1conv4w n 1conv4w m m9066 nnin b f 1.98477
1conv4w n 1conv4w m m9197 nnin b t 3.18664
1conv4w n 1conv4w m m9337 nnin b f 1.00335
1conv4w n 1conv4w m m9428 nnin b f -0.653944
1conv4w n 1conv4w m m9611 nnin b f -0.0316377
1conv4w n 1conv4w m m9816 nnin b t 6.93206
1conv4w n 1conv4w m m9833 nnin b f -0.292909
1conv4w n 1conv4w m m7066 nhpx b f 1.25729
1conv4w n 1conv4w m m7122 nhpx b f -0.605001
1conv4w n 1conv4w m m7203 nhpx b f -1.96832
1conv4w n 1conv4w m m7793 nhpx b f 1.57407
1conv4w n 1conv4w m m8363 nhpx b t 3.5626
1conv4w n 1conv4w m m8585 nhpx b f 0.192429
1conv4w n 1conv4w m m8741 nhpx b f 0.0302493
```

Figura 37. *Ejemplo del formato en el que se presentar los resultados al NIST. Dicho fichero ha sido obtenido en las simulaciones correspondientes a FA.*

En ella se puede apreciar cómo en dicho fichero hay que especificar la condición de entrenamiento y test (primera y tercera columna, respectivamente). La opción n es para especificar que no se está usando ninguna adaptación. Mediante la letra m se indica que se trata de un juicio de hombres, seguido del identificador del fichero de entrenamiento y el nombre asociado a una locución. Finalmente, se añade una t o f para definir si ambas locuciones son pronunciadas o no por la misma persona, seguido de la puntuación que acompaña a dicho juicio. Para decidirse, es necesario establecer un umbral. El umbral con el que se ha trabajado en este proyecto es de 2.5. De cualquier manera, como ya se explicó en capítulos anteriores, de cara a la EER, minDCF y DET_curve, el valor del umbral establecido es insignificante, puesto que dichos parámetros analizan las puntuaciones para un amplio rango de umbrales posibles.

- *Postnist.pl.* Genera la EER y minDCF, respectivamente. Para ello, es necesario disponer de la base de datos con los resultados correctos del año 2005 enviados por el NIST.

Finalmente, mostraremos los paquetes o funciones que hemos necesitado para las cuatro técnicas específicas que se han analizado e implementado en este proyecto. Algunas de dichas funciones son comunas para las cuatro técnicas, como la función que genera el UBM, mientras que otras son específicas de una técnica en concreto:

- *TrainWorld.* Crea el UBM, necesario en las cuatro técnicas.

- *TrainTarget*. Genera el modelo de un locutor mediante adaptación MAP (a través del algoritmo EM) o bien crea el modelo de un individuo como la suma de tres componentes independientes característico de FA. La elección de una de las dos posibilidades anteriores se especifican mediante un determinado fichero de configuración. Al igual que con la función anterior, todas las técnicas hacen uso de ésta, puesto que de cualquier forma en todas ellas se precisa de un modelo GMM para cada usuario.

- *ComputeTest*. Obtiene una puntuación para un juicio a través de la función de probabilidad condicionada característica de GMM-UBM.

- *ModelToSv*. Dado un modelo GMM creado a través de *TrainTarget*, crea el supervector correspondiente concatenando las medias.

- *Svm*. En máquinas de vectores, en función del fichero de configuración, sirve para general el hiperplano de separación o para puntuar (midiendo la distancia a dicho hiperplano).

- *naptraining.m.* ALIZE también permite trabajar con Matlab. Mediante esta función codificada en Matlab, se genera los autovectores de NAP para crear la matriz de proyección.

- *naptransform.m*. Utilizada en la técnica de NAP, proyecta un supervector a otro subespacio vectorial a través de la matriz de proyección calculada con la función anterior.

- *ComputeJFAStats.* Función que calcula una serie de estadísticos a partir del fichero con diversas grabaciones de locutores. Dichos estadísticos son necesarios para el algoritmo en el que generamos la matriz $\boldsymbol{U}$ para FA.

- *EigenChannel*. Genera $\boldsymbol{U}$.