2 SISTEMAS DE VERIFICACIÓN AUTOMÁTICA DE LOCUTOR

En este capítulo estamos interesados en abordar todos los aspectos generales de un sistema de Verificación Automática de Locutor, que nos permitirá proporcionar una base sólida para entender el resto de capítulos de este proyecto.

2.1 Verificación vs identificación

La voz almacena diferentes niveles de información. Por una parte, una señal de voz proporciona información sobre el contenido del mensaje transmitido, pero también sobre la identidad del hablante. En el primer caso nos movemos dentro del área del reconocimiento de voz (extraer el mensaje pronunciado por una persona), mientras que en el segundo trabajamos en el campo del reconocimiento de locutor (averiguar la persona que ha pronunciado un mensaje). Como ya se avanzó en la introducción, este proyecto se centra en la segunda opción.

Dentro del reconocimiento de locutor, en función de la aplicación en la que nos encontremos, podemos tener dos tareas diferentes: identificación o verificación.

- **Modo identificación (comparación 1:S).** A partir de un segmento de voz desconocido tenemos que intentar descubrir qué persona es la que está hablando. Es decir, la señal de audio de entrada se enfrenta a S posibles modelos de interlocutores. De esta forma se obtienen S puntuaciones (1 por modelo). Las puntuaciones serán más altas cuando mayor sea la probabilidad de que el segmento de voz pertenezca al modelo del hablante correspondiente. La salida del sistema será el locutor cuyo modelo obtenga la máxima puntuación. En la figura 1a) se ilustra el esquema básico de un sistema de identificación.

- **Modo verificación o detección (comparación 1:1).** En este caso el objetivo es determinar si una persona es quién dice ser. Para ello el segmento de voz de entrada se enfrenta al modelo del interlocutor que se desea comprobar y a un modelo universal que representa al resto de hablantes (impostores). La diferencia en las puntuaciones con respecto a los dos modelos se comparan con un umbral, respondiendo al sistema con un si/no (aceptar o rechazar al locutor) en función de si el resultado está por encima o por debajo de ese umbral, respectivamente. Por lo tanto un sistema de verificación se puede ver como un caso particular de un sistema de identificación cuando a este último se le agrega un modelo que representa la alternativa a todos los interlocutores o usuarios conocidos por el sistema.

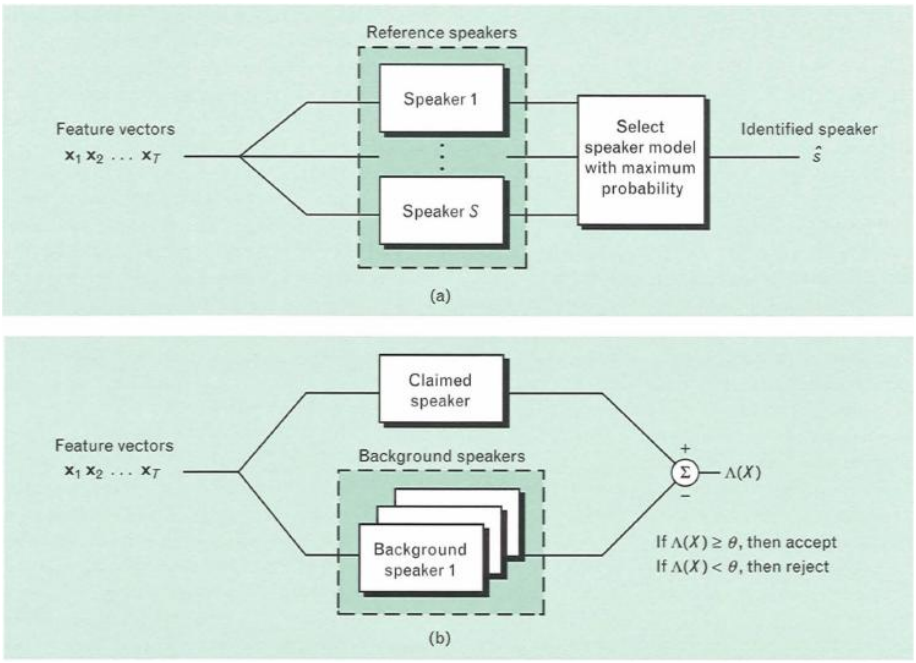

Figura 1. *Modos de funcionamiento de un sistema de reconocimiento automático de locutor. a) Identificación y b) Verficación. Figura adaptada de [D.A. Reynolds, 2008]*

Este proyecto únicamente cubrirá el área de la Verificación Automática de Locutor. Sin embargo, dentro de ésta, aún tenemos otra subdivsión: texto dependiente o texto independiente. Los sistemas de texto dependiente requieren que los usuarios pronuncien una palabra o frase determinada por el sistema, mientras que los de texto independiente están preparados para realizar el proceso de verificación independientemente de la frase pronunciada por un usuario. En este proyecto nos centraremos en el último caso.

Por lo tanto, siguiendo el criterio de clasificación que se ha llevado a cabo hasta ahora, podemos puntualizar que dentro del campo del reconocimiento de locutor, el proyecto se va a centrar en la rama de verificación con texto independiente.

2.2 Entrenamiento y test

En un sistema de verificación de locutor (y en general, en cualquier sistema de reconocimiento de voz), siempre nos vamos a encontrar con dos fases o etapas:

- **Entrenamiento.** Para poder comprobar si una persona es quién dice ser, previamente deberíamos de haber obtenido un extracto de voz de esa persona y haber creado su correspondiente modelo. En la siguiente figura podemos observar de manera general en qué consiste la fase de entrenamiento. Dado un extracto de voz de un usuario, mediante una serie de pasos (que explicaremos

detenidamente en el siguiente capítulo) obtenemos sus correspondientes parámetros característicos. Con ellos, a través de un análisis estadístico, generamos un modelo para dicho cliente, el cual será utilizado en la etapa de evaluación.

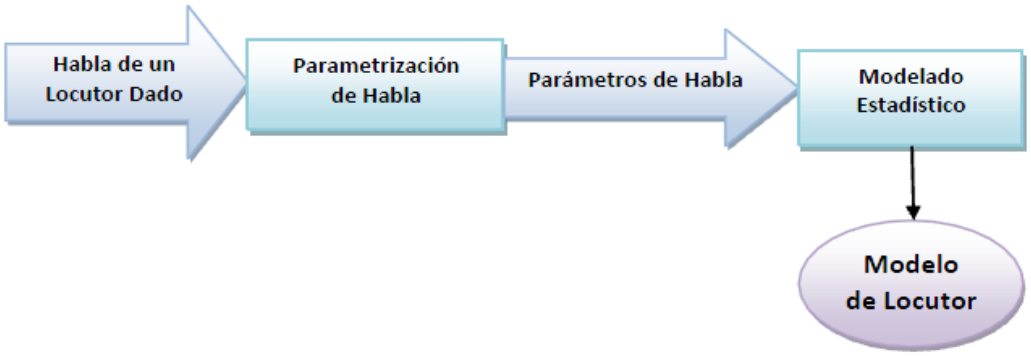

Figura 2. *Diagrama de bloques de la fase de entrenamiento de un sistema de verificación automática de locutor. Figura adaptada de [Frédéric Bimbot et al., 2004]*

- **Test (Evaluación).** Es la fase en la que un usuario quiere intentar acceder al sistema. Para comprobar si esta persona está o no registrada, necesitamos grabar su voz y obtener los parámetros correspondientes. Comparando con el modelo que se generó en la fase de entrenamiento se obtendrá una puntuación. Es muy importante aclarar que los extractos de voz utilizados para el test y el entrenamiento son diferentes (aunque naturalmente pueden provenir de la misma persona). De hecho, en una aplicación real dichos extractos serán obtenidos en diferentes instantes de tiempo. Tal y como hemos ilustrado en la figura 3, además de enfrentar el segmento de test con el modelo del locutor que se generó en la fase de entrenamiento, también se compara con un modelo que representa al resto de la población (impostores en ese determinado test). Si la persona que está siendo analizada en esta fase es quién dice ser, teoréticamente obtendrá una puntuación alta cuando se compara con el modelo obtenido en el entrenamiento, y una relativamente baja cuando se compara con el resto de la población, con lo cual la resta entre ambas puntuaciones será alta y su petición será aceptada.

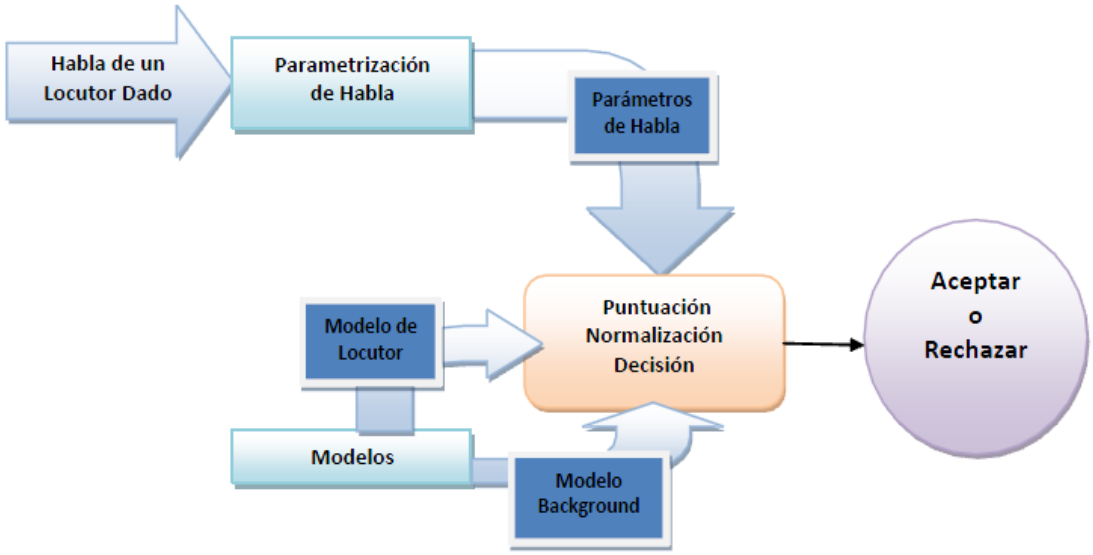

Figura 3. *Diagrama de bloques de la fase de test de un sistema de verificación automática de locutor. Figura adaptada de [Frédéric et al., 2004]*

De acuerdo con las características de un sistema de verificación, la idea clave y general a la que nos vamos a enfrentar en todo el proyecto es la siguiente: a partir de un segmento de voz de entrenamiento y otro de test, ¿podemos afirmar que ambos extractos pertenecen a la misma persona? De ahí el carácter binario de un sistema de verificación de voz, cuya respuesta simplemente consistirá en aceptar o rechazar a un determinado cliente (sí o no) y que se puede visualizar en la siguiente figura:

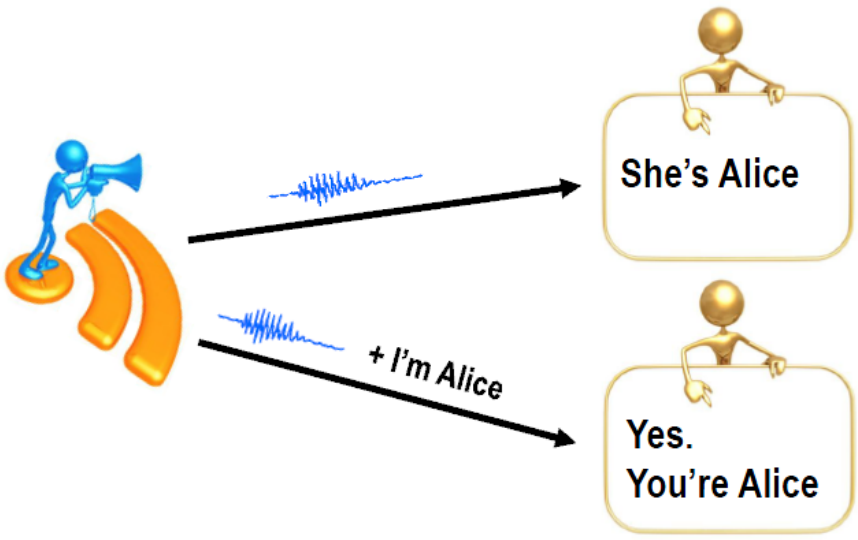

Figura 4. *Ejemplo para ilustrar las dos fases de un sistema de verificación automática de locutor*

Tal y como se ha mencionado anteriormente y se puede apreciar perfectamente en la figura 3, un sistema de verificación de voz representa un sistema completamente

estadístico. Dado dos segmentos de voz $\boldsymbol{X}$ e $\boldsymbol{Y}$, el objetivo es encontrar cuál de las dos siguientes hipótesis es la más probable:

- H_0: los segmentos $\boldsymbol{X}$ e $\boldsymbol{Y}$ fueron hablados por el mismo locutor.
- H_1: los segmentos $\boldsymbol{X}$ e $\boldsymbol{Y}$ no fueron hablados por el mismo locutor (hipótesis alternativa).

2.3 Medidas de rendimiento y presentación de resultados

En el presente proyecto se necesitarán una serie de parámetros para evaluar y poder comparar los resultados de las diferentes técnicas que se implementen. Tal y como se añadió en la introducción, las simulaciones que se van a llevar a cabo durante el proyecto son con datos del NIST. Aunque en capítulos posteriores analizaremos con mayor detalle toda la información relativa a este organismo, sería bastante aconsejable adelantar que básicamente éste te manda un conjunto de segmentos de audio de entrenamiento y de evaluación. A partir de estos datos, te propone una sería de juicios ("*trials"*) que consisten en pares formados por un fichero de entrenamiento y otro de test. El objetivo es averiguar si dichos pares pertenecen o no a la misma persona. Luego en nuestro sistema definiremos un determinado umbral, y en función de si la puntuación obtenida en un juicio cae por encima o por debajo de éste, concluiremos que el juicio tiene un resultado positivo o negativo, respectivamente.

2.3.1 Probabilidad de falsa alarma y de falso rechazo

En cualquier sistema de verificación hay dos tipos de errores. Dado dos segmentos de audio (uno de entrenamiento y otro de evaluación) que pertenezcan a diferentes interlocutores ("*impostor trial")*, un primer tipo de error que se puede cometer es considerar que dichos ficheros pertenecen a la misma persona. A este fallo se le conoce como error de falsa alarma o de falsa aceptación y en la nomenclatura del NIST y en el proyecto lo vamos a representar mediante "P_{fa}" (expresado en % indicando el porcentaje de fallos que se ha cometido dentro de los juicios de impostores). Por otra parte, dado dos ficheros que pertenezcan a la misma persona ("*true trial*"), un segundo tipo de error consiste en afirmar que dichos segmentos son pronunciados por diferentes locutores. A este error se le conoce como error de falso rechazo o de pérdida y se representa mediante "P_{miss}".

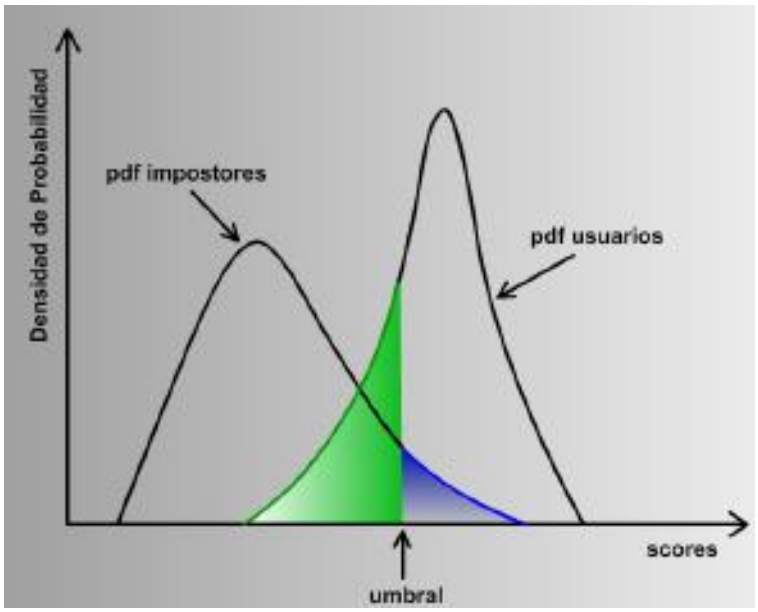

Figura 5. *Curvas de densidad de probabilidad de las puntuaciones de usuarios e impostores*

En la figura 5 se puede observar claramente el motivo por el cual cometemos estos tipos de errores. En ella se ha representado la función densidad de probabilidad de las puntuaciones ("*scores*") obtenidas con los juicios de usuarios ("*target*") y de impostores ("*non-target*"). El área de color verde bajo la curva de usuarios representa la probabilidad de falso rechazo, mientras que la de azul bajo la curva de impostores representa la probabilidad de falsa aceptación. Analizando la gráfica se puede apreciar que un parámetro fundamental es la ubicación del umbral. Si lo incrementamos, aumentaremos la probabilidad de falso rechazo y disminuiremos la de falsa alarma y viceversa. Por consiguiente, el valor del umbral determinará el punto de trabajo de nuestro sistema. En aplicaciones de seguridad, por ejemplo, donde se tiene que evitar a toda consta que los impostores puedan entrar en el sistema, se trabajan con valores altos del umbral, con el inconveniente de que se incrementará la probabilidad de error para los usuarios.

2.3.2 Curvas ROC y DEC

Tal y como se ha explicado en relación a la figura 5, en función del umbral establecido, se obtiene un determinado valor de P_{fa} y otro de P_{miss}. Así que disponemos de infinitos puntos de trabajo, donde cada uno de ellos vendrá determinado por un umbral en concreto. Representando en el eje de abscisas la probabilidad de falsa alarma y en el eje de ordenadas la de falso rechazo, para todos los valores del umbral posibles se obtiene lo que se conoce como curva ROC (Receiver Operating Characteristic). De esta forma se muestra la precisión del sistema en todos los puntos de trabajo.

Cuando dibujamos en el mismo gráfico varias curvas ROC de distintos sistemas de verificación con rendimientos parecidos, estas tienden a quedar muy próximas, tanto que puede llegar a resultar difícil diferenciarlas. Para evitar este problema, [Martin and Przybocki, 2004] introdujeron una variante de la curva ROC conocida como DET (Detection Error Trade-Off), obtenida realizando una transformación de desviación normal sobre los ejes de la primera. Esta transformación está motivada por el hecho de que si las distribuciones de probabilidad de impostores y usuarios siguen aproximadamente una distribución normal (condición suficiente, pero no necesaria) las curvas DET producidas serán líneas rectas.

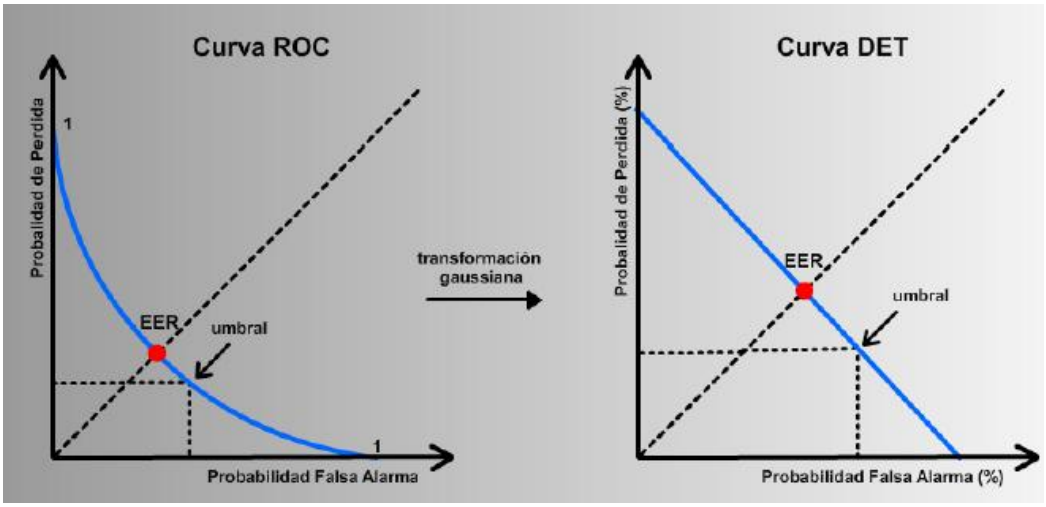

Figura 6. *Curvas ROC y DET*

2.3.3 EER (Equal Error Rate)

Un parámetro totalmente neutro e imparcial para analizar el rendimiento de un sistema de verificación de locutor es la EER. En la gráfica anterior, podemos ver que dicho parámetro se corresponde con la intersección de la curva DET con la recta $P_{fa}=P_{miss}$. Es decir, la EER coincide con el porcentaje de error de falsa alarma o de falso rechazo cuando estos dos tipos de errores coinciden. De cara a una aplicación real, este parámetro carece de relativa importancia puesto que en muy pocos casos estaremos interesados en que la probabilidad de falsa alarma y de falso rechazo coincida (generalmente se penaliza en mayor medida al error de falsa alarma que al de rechazo). Sin embargo, ha sido ampliamente utilizado en este proyecto debido a su alto grado de neutralidad y objetividad. Lógicamente, cuanto más pequeño sea el valor de la EER tanto mayor será la precisión de un sistema de verificación.

2.3.4 ***DCF (Detection Cost Function)***

Este es el parámetro que utiliza el NIST [Martin and Przybocki, 2004] para evaluar los resultados de los participantes en cualquier evaluación. Con él, el NIST intenta analizar los resultados desde un punto de vista más realista y más adaptado a lo que sería en una aplicación real. Para ello le asocia a los dos tipos de errores que se pueden cometer un determinado coste:

$$C = P \times C_{miss} \times P_{miss} + (1 - P) \times C_{fa} \times P_{fa}, \qquad (1.1)$$

donde C_{miss} y C_{fa} representan los costes de los errores de pérdida y falsa alarma, respectivamente. P es la probabilidad a priori de tener un juicio en el que los dos ficheros o segmentos de voz pertenecen al mismo locutor ("*true trial*"), mientras que P_{miss} y P_{fa} hacen referencia a las probabilidades de pérdida y falsa alarma, respectivamente. De esta forma, el NIST penaliza de una forma diferente un error de falsa alarma (C_{fa}) que un error de pérdida (C_{miss}), asemejándose a lo que probablemente nos encontraríamos en una aplicación real.

Tal y como hemos verificado anteriormente, P_{miss} y P_{fa} dependen del umbral que definamos (todas las posibilidades se pueden contemplar observando la curva DET, por ejemplo). Por lo tanto, de todas las posibilidades existentes, se escogerán aquellos valores de P_{miss} y P_{fa} que minimicen la función de coste C, y a dicho valor de coste se le conoce como minDCF. El participante que obtenga el minDCF más bajo se convertirá en el ganador de la correspondiente evaluación del NIST.

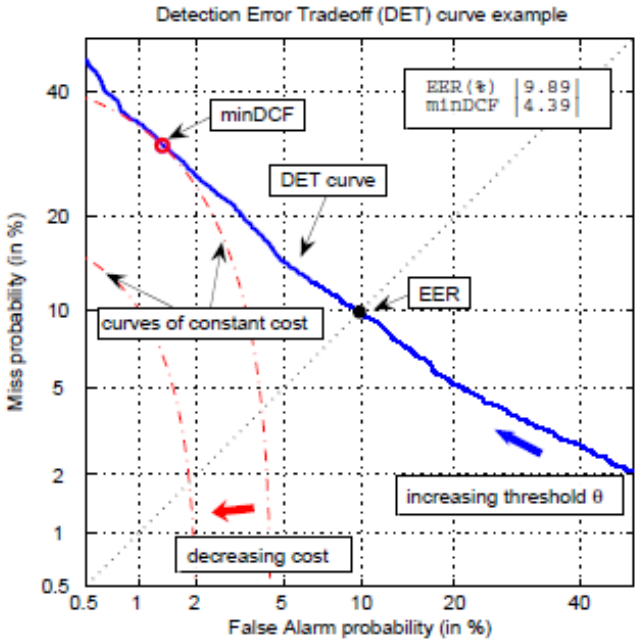

Figura 7 *Ejemplo de una curva DET con los correspondientes valores de EER y minDCF*

La figura 7 identifica los tres tipos de parámetros que vamos a utilizar a lo largo de este proyecto para analizar la eficiencia de los diferentes sistemas y compararlos entre ellos: curva DET, EER y minDCF. En ella podemos ver cómo al aumentar el valor del umbral nos movemos hacia la parte superior izquierda de la gráfica, es decir, hacia zonas donde la probabilidad de falsa alarma es baja y la de pérdida es relativamente alta. También vemos como el punto rojo representa el mínimo valor de la función de coste (minDCF). De igual manera, sobre dicha figura se han representado en líneas rojas discontinuas curvas de coste constante.

Los valores asignados por el NIST son los siguientes:

$$P = 0.01 \quad C_{miss} = 10 \quad C_{fa} = 1$$

A simple vista, podríamos predecir que puesto que el coste de pérdida o falso rechazo es diez veces más grande que el de falsa alarma, el mínimo de la función de coste debería de situarse en la parte derecha de una curva DET (alto P_{fa} y bajo P_{miss}). Sin embargo, hay que tener en cuenta que la probabilidad de encontrarse un examen de usuarios ("*true trial*") es cien veces más pequeña que la de encontrarse un juicio de impostores, por lo que el producto $P \times C_{miss}$ es más bajo que $(1 - P) \times C_{fa}$, concluyendo por tanto que la penalización es mayor cuando nos equivocamos en un juicio de impostores (para alcanzar el minDCF hay que intentar minimizar P_{fa}). Esto justifica que en el caso concreto del NIST, el minDCF se corresponderá con un punto situado en la parte superior izquierda de la curva DET, tal y como sucede en la figura 7.

De acuerdo con lo anterior, minimizar la función de coste nos lleva a trabajar en puntos con probabilidad de falsa alarma menor que de falso rechazo. Asemejando esta idea con una aplicación real como una transacción bancaria, por ejemplo, esto se traduce en desarrollar un sistema que intente reducir al máximo la probabilidad de que un impostor pueda acceder a una cuenta de la que no es propietario, y no otorgue tanta prioridad a que un usuario registrado no pueda acceder. Si un cliente no consigue entrar, pues tendrá la posibilidad de un segundo intento para enmendar el error del sistema. Sin embargo, el coste o la penalización de permitir a un impostor acceder a la cuenta son enormes. Así que el criterio que sigue el NIST para evaluar los sistemas de verificación concuerda bastante con lo que nos podríamos encontrar en la realidad.

Finalmente, comentar que además de la curva DET, los dos criterios numéricos que se utilizarán en este proyecto para evaluar los resultados de las simulaciones son el minDCF y la EER. Está claro que teóricamente valores pequeños de la EER deberían de corresponderse con valores bajos del minDCF. Sin embargo, la relación entre estos dos parámetros no es de ninguna manera lineal, con lo que para sistemas con rendimiento muy similares, es posible encontrarse con un cierto grado de ambigüedad en relación a estos dos parámetros. En ese caso, siempre priorizaremos con el minDCF, por ser el parámetro de evaluación que utiliza el NIST.