

Indexación de audio: Reconocimiento de Locutor e Idioma

<audias>

Audio, Data Intelligence and Speech

<http://audias.ii.uam.es>

Daniel Ramos Castro

Coautores: Doretoe Torre, Joaquín González, Alicia Lorenzo

0

Contenidos

- **Introducción**
- **Reconocimiento del locutor independiente del texto**
- **Reconocimiento del locutor dependiente del texto**
- **Reconocimiento del idioma**

El problema (ejemplo en ciencia forense)

- Grabación incriminatoria (**dubitada**)
 - ▣ Pinchazo telefónico
 - ▣ Llamada anónima
 - ▣ Micrófono oculto
 - ▣ ...
- La policía arresta a un sospechoso
- Se realiza una toma de voz del sospechoso (**indubitada**)
 - ▣ En dependencias policiales
 - ▣ Pinchazos cuya autoría se reconoce
 - ▣ ...
- El contenido lingüístico no se conoce a priori en ambos casos
 - ▣ Independiente de texto



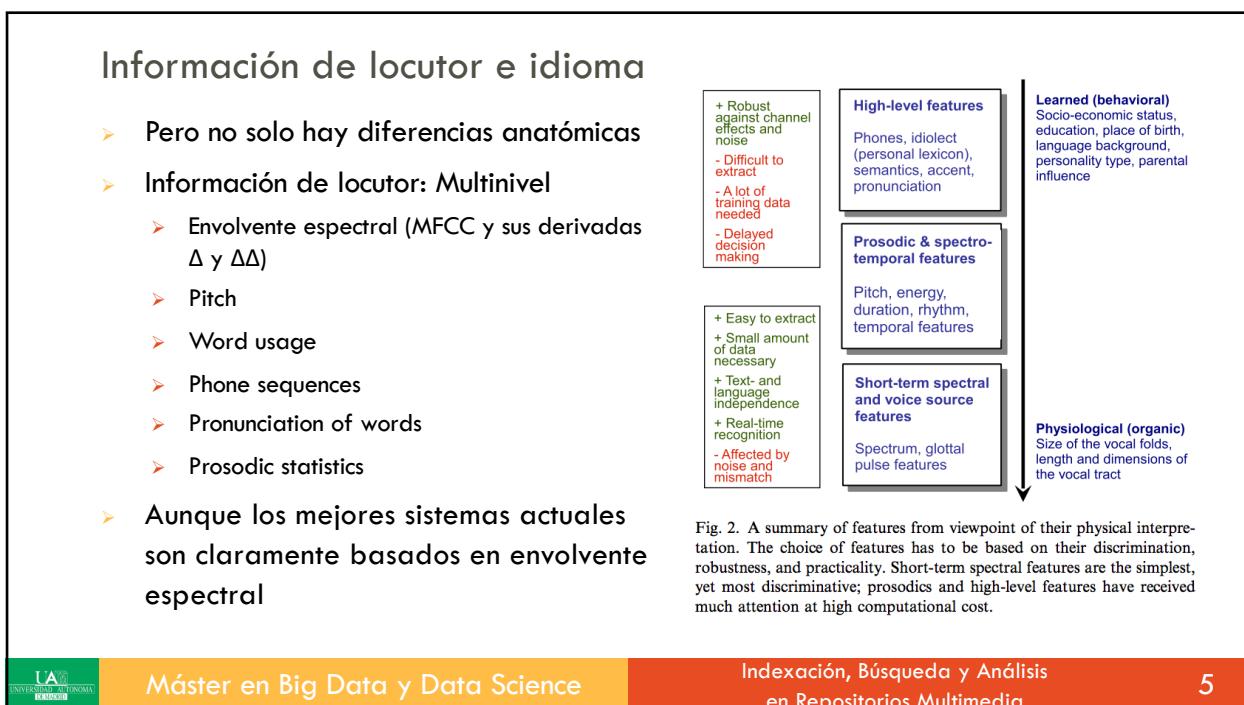
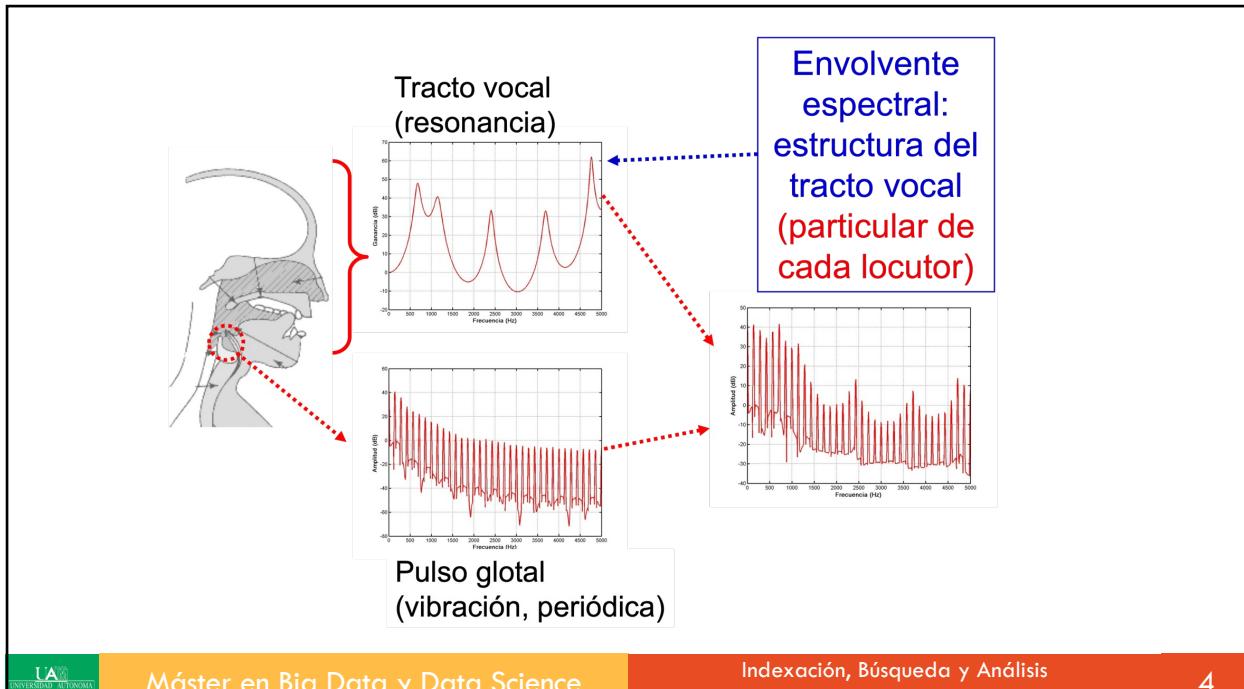
**Criminal
(Identidad C)**



**Sospechoso
(Identidad S)**

Reconocimiento automático de locutores

- Principio de funcionamiento
 - ▣ Dos personas tienen tractos vocales diferentes
 - Porque son anatómicamente diferentes
 - ▣ Por tanto, estas diferencias se tienen que reflejar en la envolvente espectral del sonido, representado por los MFCC (por ejemplo)
 - ▣ Por tanto, podemos distinguir locutores mediante dichas características acústicas



Reconocimiento automático de locutores

- La gran mayoría de sistemas calcula puntuaciones (**scores**)
- Similitud entre las **identidades** en dos fragmentos de voz



- Idealmente:
 - Si C y S son la misma identidad, score más alto
 - Si C y S son identidades diferentes, score más bajo
- Un score permite **discriminar**

Reconocimiento de Locutores: Dependencia del Texto

- Sistemas dependientes de texto
 - El contenido lingüístico de lo que se va a reconocer es conocido
 - Ejemplo: "diga su PIN"
 - Ejemplo: "diga la siguiente frase" (*text-prompted*)
 - Aplicación típica: acceso a cuenta bancaria
- Independiente de texto
 - El contenido lingüístico no se conoce a priori (podría decirse cualquier cosa)
 - Aplicaciones típicamente de inteligencia y forenses
 - Ejemplos:
 - Búsqueda de voz en bases de datos de audio de sospechosos
 - Localización de un sospechoso en llamadas
 - Presentación de evidencias de voz en juicios



Reconocimiento de Locutores: Modos de Funcionamiento

□ Modo de verificación o autenticación de locutores

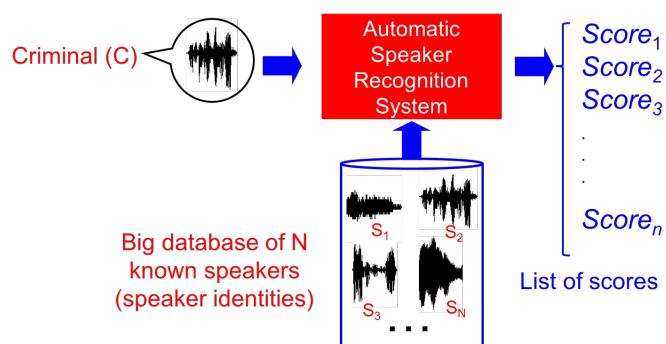
- ❑ El score se utiliza para tomar una decisión sobre si la persona que está intentando acceder al sistema es quien dice ser
- ❑ Existe una identidad supuesta de ese locutor
- ❑ Se verifica si es él o no
- ❑ Modo típico en aplicaciones de control de acceso



Reconocimiento de Locutores: Modos de Funcionamiento

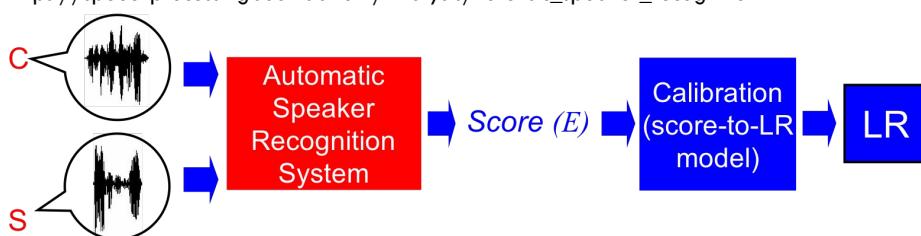
□ Modo de identificación de locutores

- ❑ El score se utiliza para buscar en una base de datos
 - ❑ Muchos scores, de muchas comparaciones
- ❑ Se genera un ranking
- ❑ Aplicación típica: búsqueda de una persona en una base de datos de locutores



Reconocimiento de Locutores: Modos de Funcionamiento

- Modo de **evaluación forense de evidencias** de voz
 - El score se utiliza como evidencia en un juicio
 - Se evalúa el peso de dicha evidencia
 - Ese peso se expresa como un ratio de verosimilitudes (likelihood ratio, LR)
 - Aplicación típica: valoración de pruebas de voz en juicios (ciencia forense)
 - Detalles en el siguiente enlace (**no evaluable en examen**)
 - https://speechprocessingbook.aalto.fi/Analysis/Forensic_speaker_recognition.html



Cálculo de una puntuación (score): etapas

- Enfoque clásico:
- Paso 1: Modelado de características

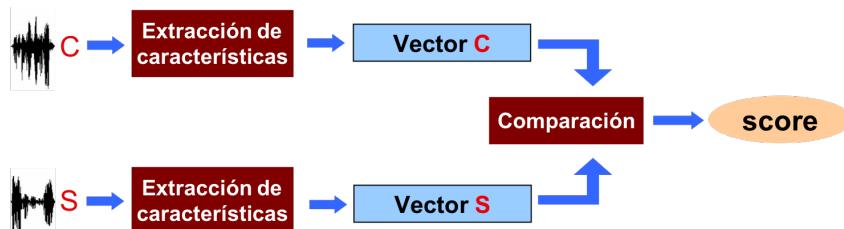


- Paso 2: Cálculo de puntuación (score) entre modelo y características



Cálculo de una puntuación (score): etapas

- Enfoque más moderno: sustituye el paso 2 por la comparación de vectores que representan al locutor
 - ▣ Representaciones vectoriales de cada audio a comparar
 - ▣ Pueden extraerse mediante modelado estadístico (i-Vectors) o mediante DNNs (x-Vectors)



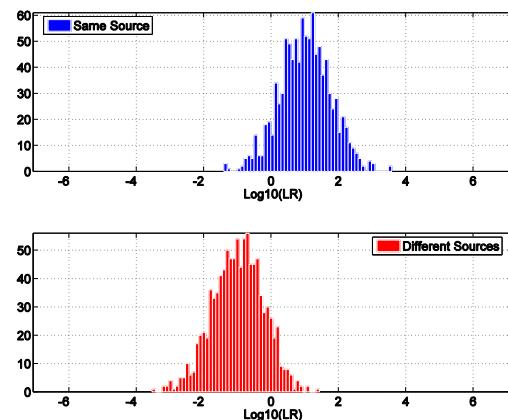
Evaluación Empírica de Sistemas de Reconocimiento de Locutores

- Requerimiento fundamental de cualquier tecnología
- Utilización de una base de datos de locuciones
 - ▣ Sabemos a qué locutor pertenece cada toma
 - Dubitada
 - Indubitada
 - ▣ Por tanto, sabemos las “respuestas correctas” (*ground-truth*)
- Hacemos muchas comparaciones diferentes: generamos muchos valores de LR
 - ▣ Podemos separar los scores obtenidos dependiendo de la fuente de los dos audios
 - Scores que provienen del mismo locutor (*misma fuente, “target”*)
 - Scores que provienen de dos locutores distintos (*fuentes diferentes, “non-target”*)

¿Criterio de bondad?

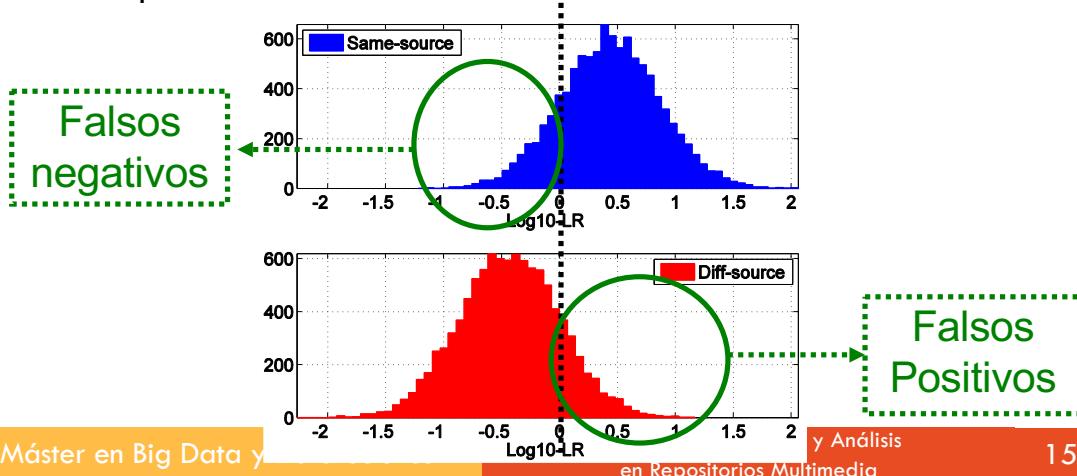
- La separación entre ambos tipos de score es deseable

- Cuanto más separados, mejor distingue el método de evaluación entre casos en los que cada proposición es cierta



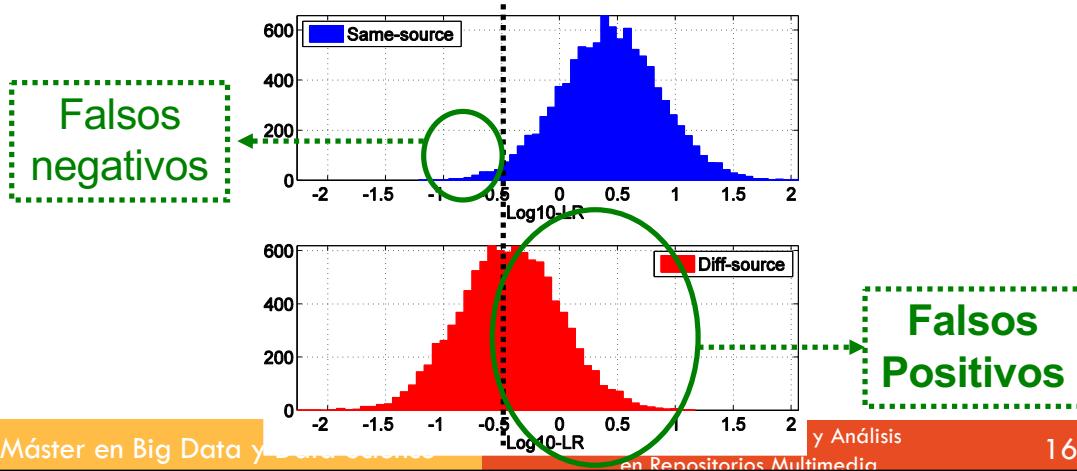
Falsos Positivos y Falsos Negativos

- Medida de rendimiento clásica
- Se miden para valores concretos de los umbrales de decisión
- Medida del poder de discriminación



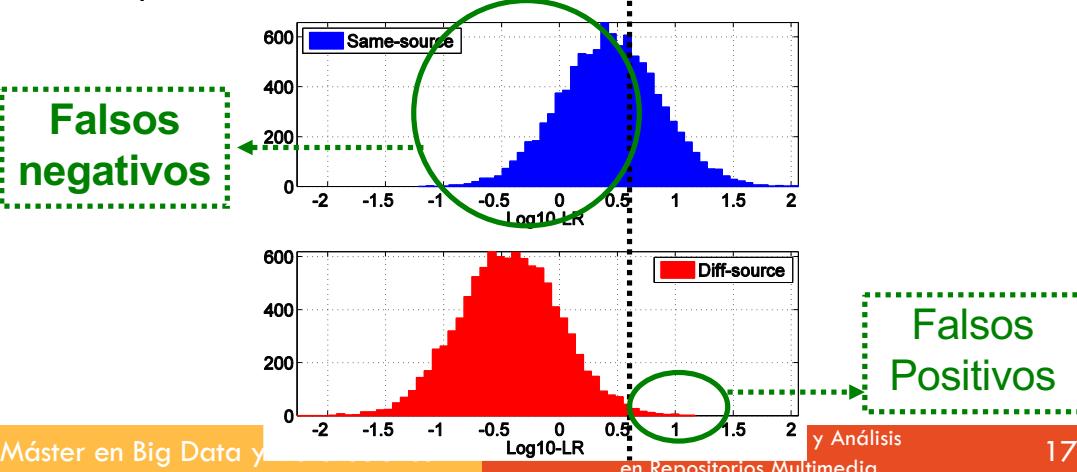
Falsos Positivos y Falsos Negativos

- Medida de rendimiento clásica
- Se miden para valores concretos de los umbrales de decisión
- Medida del poder de discriminación



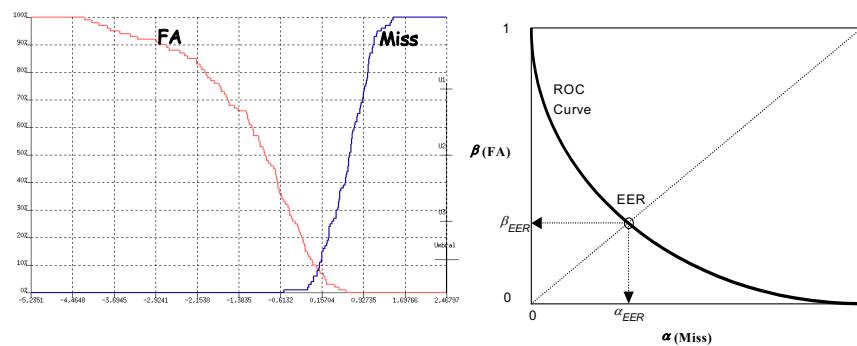
Falsos Positivos y Falsos Negativos

- Medida de rendimiento clásica
- Se miden para valores concretos de los umbrales de decisión
- Medida del poder de discriminación



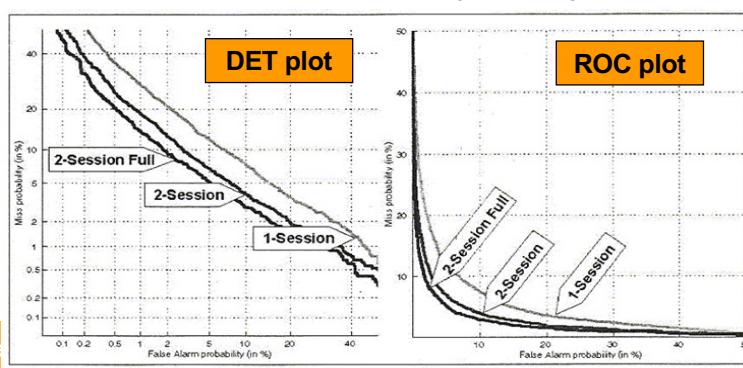
Curvas ROC

- ROC (Receiver Operating Curve): Falsos negativos (o falsos rechazos, miss) vs. Falsos Positivos (o falsas aceptaciones, falsas alarmas)
 - Para cualquier umbral Th (score S_{Th})
 - Dibujar puntos de (FA_{Th} , $Miss_{Th}$)



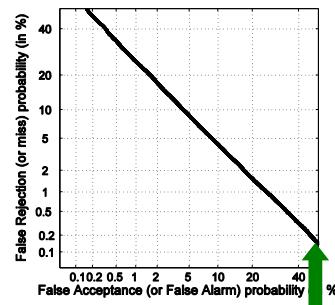
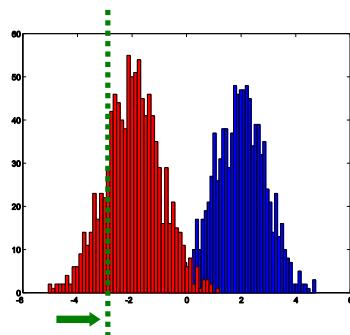
Curvas DET

- Las mismas probabilidades que la curva ROC
 - Pero en una escala gaussianizada
 - Distribuciones Gaussianas serían rectas
- Permite una visualización mejor a bajas tasas de error



Curvas DET

- Todos los puntos de trabajo (umbrales) se visualizan en una sola gráfica

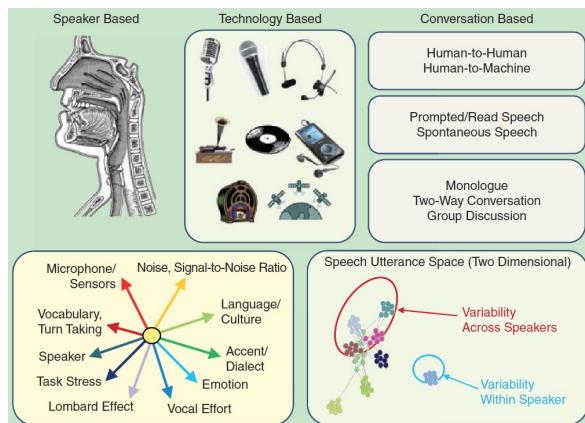


Variabilidad interlocutor e intersetión

- **Variabilidad interlocutor**
 - Variabilidad entre diferentes locutores
 - Por el hecho de ser personas diferentes
 - Es la variabilidad que pretende capturar un sistema de reconocimiento de locutores
- **Variabilidad intersetión**
 - Diferencias acústicas existentes entre sesiones de un mismo locutor
 - Diferentes locuciones, potencialmente en diferentes sesiones de grabación, tienen diferencias acústicas
 - Hace que dos locutores iguales parezcan diferentes, degradando el rendimiento de los sistemas
 - Variabilidad **no deseada** en sistemas de reconocimiento de locutor
- **La variabilidad intersetión provoca desajustes de condiciones**
 - Entre las dos locuciones a comparar
 - Entre la voz que entrena al sistema de reconocimiento y las voces a comparar
 - **Estos desajustes hacen que los sistemas reconozcan mucho peor**

Variabilidad interlocutor e intersetión

- Múltiples posibles fuentes de variabilidad entre locutores y entre sesiones



John H. L. Hansen, Taufiq Hasan. "Speaker Recognition by Machines and Humans: A tutorial review". IEEE Signal Process. Mag. 32(6), pp. 74-99, 2015

Contenidos

- Introducción
- Reconocimiento del locutor independiente del texto
- Reconocimiento del locutor dependiente del texto
- Reconocimiento del idioma

Sistemas Clásicos GMM-UBM

- Se genera un modelo de mixturas de gaussianas para el audio S
- Adaptado desde un modelo universal (Universal Background Model, UBM) que representa al “locutor genérico”, y que se entrena con muchos locutores diferentes
- Paso 1: Modelado de características (Modelo GMM-UBM)



- Paso 2: Cálculo de puntuación (score) entre GMM-UBM de S y las características de C



GMM adaptado desde un UBM

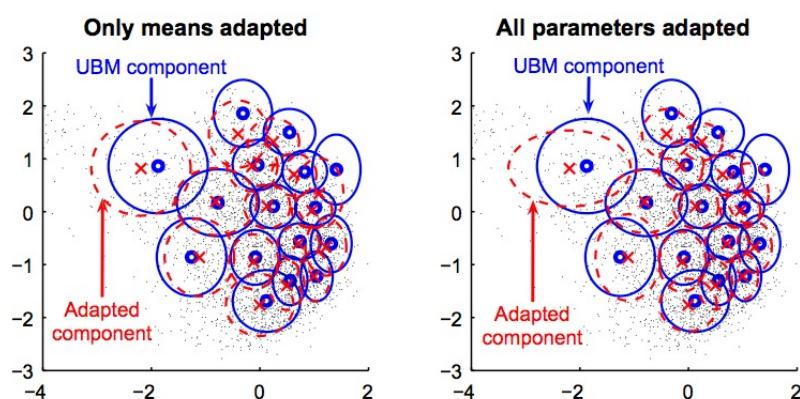
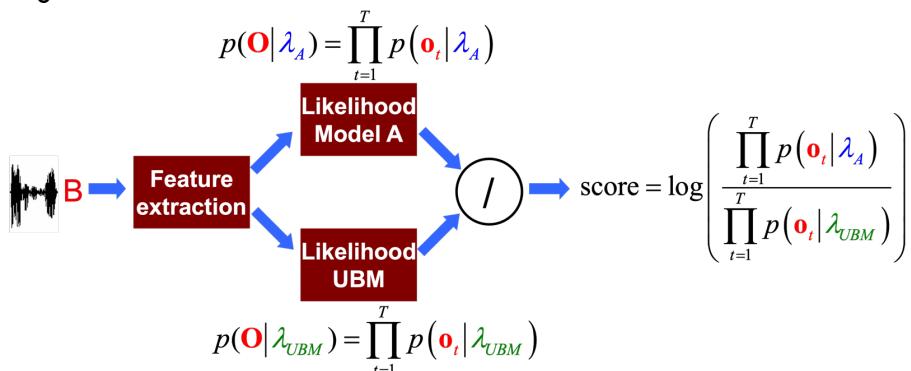


Fig. 8. Examples of GMM adaptation using *maximum a posteriori* (MAP) principle. The Gaussian components of a universal background model (solid ellipses) are adapted to the target speaker's training data (dots) to create speaker model (dashed ellipses).

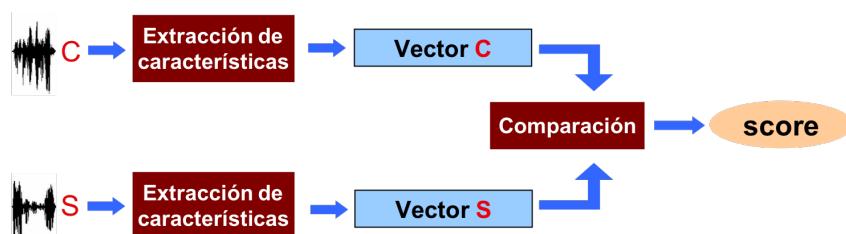
Cálculo de la Puntuación (score) con GMM-UBM

- Observaciones (MFCC) del locutor B: \mathbf{o}
- Modelo GMM del locutor A (adaptado del UBM): λ_A
- Modelo UBM (universo de locutores): λ_{UBM}
- Score: log-ratio de verosimilitudes



Sistemas Modernos (comparación de vectores)

- Enfoque más moderno: sustituye el paso 2 por la comparación de vectores que representan un audio a comparar
 - ▣ Pueden extraerse mediante modelado estadístico (i-Vectors)
 - ▣ O mediante DNNs que extraen embeddings (representaciones vectoriales de un audio, como por ejemplo los llamados x-Vectors)
 - ▣ Pero el mecanismo de comparación es el mismo: cada vector representa un audio a comparar

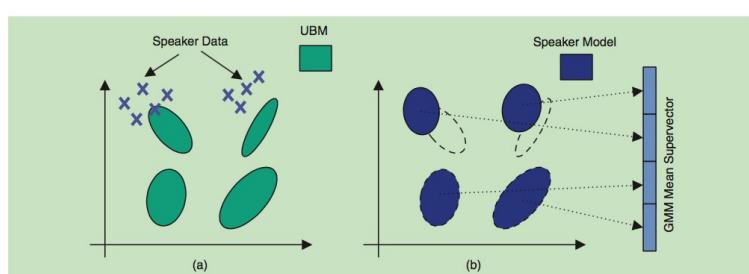


i-Vector

- Vector que representa un fragmento de audio (uno o varios ficheros)
- Se obtiene desde el GMM-UBM entrenado con un audio
- Se aplican técnicas de subespacios vectoriales
 - ▣ Básicamente:
 - Se convierte un GMM-UBM en un vector de muchas dimensiones (supervector GMM)
 - Concatenando las medias de las gaussianas del GMM-UBM
 - O calculando estadísticos del audio a representar con el UBM
 - Se obtiene el subespacio en el que varían las locuciones de un corpus
 - Utilizando técnicas de análisis factorial
 - Se extrae un i-Vector por cada locución a comparar, y se comparan dichos i-Vectors
 - De este modo, un i-Vector es un vector que está en el subespacio de un espacio de GMM (o más bien de sus supervectores) donde se ubican realmente los audios

i-Vector

- Primer paso: transformación de un GMM-UBM en un supervector
 - ▣ Concatenando estadísticos de un GMM, típicamente concatenándolos de dos formas posibles
 - O concatenando los vectores de medias del GMM-UBM adaptado
 - O usando los estadísticos de Baum-Welch de primer orden de las observaciones en el UBM



[FIG7] A schematic diagram of a GMM-UBM system using a four-mixture UBM. MAP adaptation procedure and supervector formation by concatenating the mean vectors are also illustrated. (a) A schematic diagram of a GMM-UBM system using a four-mixture UBM. (b) MAP adaptation procedure and supervector formation by concatenating the mean vectors are also illustrated.

- Segundo paso: modelado de subespacio de variabilidad total (T)
 - ▣ Para extraer i-Vectors de los audios a comparar

x-Vector

- Embeddings extraídos con una red neuronal profunda (DNN)
 - ▣ Red de extracción de embeddings:
 - DNN que intenta clasificar un conjunto cerrado de locutores
 - ▣ Las primeras capas del extractor son capas Bilateral-LSTM y Fully-Connected que trabajan a nivel de características (MFCC) x_i
 - Una característica por cada trama temporal (por cada vector MFCC)
 - ▣ Tras ello, se acumulan los estadísticos de la red en el tiempo en una capa intermedia (*Statistics Pooling*)
 - A partir de ahí el resto de capas representan la locución completa
 - Parte de esas capas forman el vector que representa al audio de entrada (*embedding*)
 - ▣ Se suelen utilizar estrategias de aumento de datos (*data augmentation*)

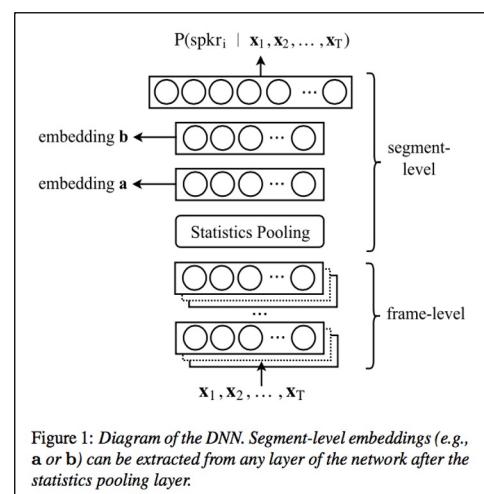


Figure 1: Diagram of the DNN. Segment-level embeddings (e.g., a or b) can be extracted from any layer of the network after the statistics pooling layer.

D Snyder et al. "X-vectors: Robust DNN embeddings for speaker recognition". ICASSP 2018.

x-Vector

- Otras arquitecturas de extracción de x-Vectors son las redes Time-Delay Neural Networks (TDNN)
 - ▣ En ocasiones incluyendo mecanismos de atención (redes ECAPA-TDNN)

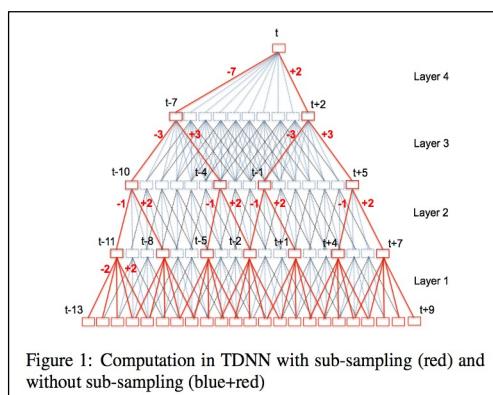


Figure 1: Computation in TDNN with sub-sampling (red) and without sub-sampling (blue+red)

Comparación de vectores de locutor (i-Vectors o x-Vectors): PLDA

- PLDA: Probabilistic Lineal Discriminant Analysis
- Score PLDA calculado a partir de dos vectores w_1 y w_2 (pueden ser x-Vectors o i-Vectors):

$$S_{w_1, w_2} = \frac{p(w_1, w_2 | H_0)}{p(w_1 | H_1)p(w_2 | H_1)} = \frac{\int p(w_1, w_2 | h)p(h)dh}{\int p(w_1 | h_1)p(h_1)dh_1 \int p(w_2 | h_2)p(h_2)dh_2}$$

- Para este cálculo se utilizan subespacios de variabilidad interlocutor e intersetión
 - El score busca representar la variabilidad interlocutor (deseada)
 - Elimina (promedia) la variabilidad entre sesiones (compensación de variabilidad intersetión)
- Se intenta que esas integrales se solucionen analíticamente
 - Simplificación del cálculo
- El score se interpreta como un ratio de verosimilitudes entre dos hipótesis
 - Hipótesis 1 (numerador): w_1 y w_2 provienen del mismo locutor
 - Hipótesis 2 (denominador): w_1 y w_2 provienen de locutores diferentes

Rendimiento de x-Vectors comparado con i-Vectors

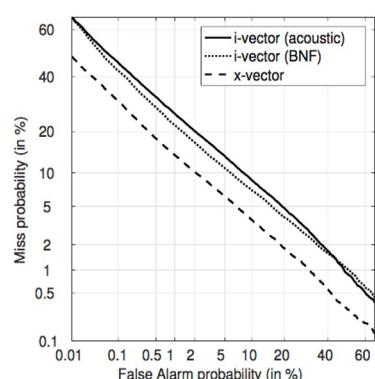


Fig. 1. DET curve for the Cantonese portion of NIST SRE16 using Section 4.5 systems.

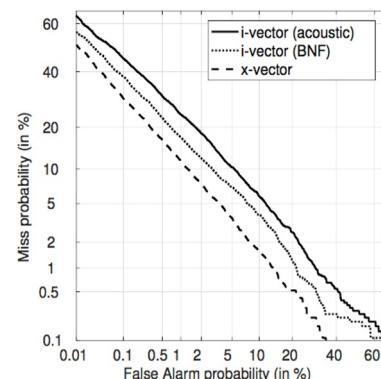


Fig. 2. DET curve for the SITW Core using Section 4.5 systems.

D Snyder et al. "X-vectors: Robust DNN embeddings for speaker recognition". ICASSP 2018.

Referencias en Reconocimiento de Locutor Independiente de Texto

Computer Speech & Language 60 (2020) 101026



State-of-the-art speaker recognition with neural network embeddings in NIST SRE18 and Speakers in the Wild evaluations



Jesús Villalba^{a,*}, Nanxin Chen^a, David Snyder^{a,b}, Daniel García-Romero^b, Alan McCree^b, Gregory Sell^b, Jonas Borgstrom^c, Leibny Paola García-Pereira^a, Fred Richardson^a, Réda Dehak^d, Pedro A. Torres-Carrasco^a, Najim Dehak^c

^a Center for Language and Speech Processing, Johns Hopkins University, Baltimore MD, USA

^b Human Language Technology Center of Excellence, Johns Hopkins University, Baltimore MD, USA

^c MIT Lincoln Laboratory, Lexington MA, USA

^d LSE-EPITA, Villejuif, France

ARTICLE INFO

Article History:
Received 5 May 2019
Revised 10 September 2019
Accepted 26 September 2019
Available online 9 October 2019

Keywords:
Speaker recognition
Embeddings
X-vectors
NIST SRE18
Domain adaptation
Evaluations
Calibration

ABSTRACT

We present a thorough analysis of the systems developed by the JHU-MIT consortium in the NIST SRE18 evaluation. In 2015, i-vectors were the speaker recognition state-of-the-art. However now, neural network embeddings (a.k.a. x-vectors) rise as the best performing approach. We show that in some conditions, x-vectors' detection error reduces by 2x w.r.t. i-vectors. In this work, we experimented on the Speakers in The Wild evaluation (STW), NIST SRE18 VAST (Video Annotation for Speech Technology), and SRE18 CMN2 (Call My Net 2, telephone Tunisian Arabic) to compare network architectures, pooling layers, training objectives, back-end adaptation methods, and calibration techniques. X-Vectors based on factorized and extended TDNN networks achieved performance without parallel on STW and CMN2 data. However for VAST, performance was severely degraded. This was due to the fact that the VAST test set was severely degraded compared to the STW, even though they both consist of internet videos. This degradation caused strong domain mismatch between training and VAST data. Due to this mismatch, large networks performed just slightly better than smaller networks. This also complicated VAST calibration. However, we managed to calibrate VAST by adapting STW scores distribution to VAST, using a small amount of in-domain development data.

Indexación, Búsqueda y Análisis
en Repositorios Multimedia

34

34

Referencias en Reconocimiento de Locutor Independiente de Texto

Forensic Science International: Synergy 4 (2022) 100223



Validations of an alpha version of the E³ Forensic Speech Science System (E³FS³) core software tools



Philip Weber^{a,b}, Ewald Enzinger^{a,b,c}, Beltrán Labrador^d, Alicia Lozano-Díez^d, Daniel Ramos^d, Joaquín González-Rodríguez^e, Geoffrey Stewart Morrison^{a,b,*}

^a Forensic Data Science Laboratory, Aston University, Birmingham, UK

^b Forensic Evaluation Ltd, Birmingham, UK

^c Edworks Corporation, Corvallis, OR, USA

^d AUDIAS – Audio, Data Intelligence and Speech, Escuela Politécnica Superior, Universidad Autónoma de Madrid, Madrid, Spain

ARTICLE INFO

Keywords:
Forensic voice comparison
Validation
Likelihood ratio
x-vector

ABSTRACT

This paper reports on validations of an alpha version of the E³ Forensic Speech Science System (E³FS³) core software tools. This is an open-code human-supervised-automatic forensic-voice-comparison system based on x-vectors extracted using a type of Deep Neural Network (DNN) known as a Residual Network (ResNet). A benchmark validation was conducted using training and test data (*forensic_eval01*) that have previously been used to assess the performance of multiple other forensic-voice-comparison systems. Performance equalled that of the best-performing system with previously published results for the *forensic_eval01* test set. The system was then validated using two different populations (male speakers of Australian English and female speakers of Australian English) under conditions reflecting those of a particular case to which it was to be applied. The conditions included three different sets of codecs applied to the questioned-speaker recordings (two mismatched with the set of codecs applied to the known-speaker recordings), and multiple different durations of questioned-speaker recordings. Validations were conducted and reported in accordance with the "Consensus on validation of forensic voice comparison".

Indexación, Búsqueda y Análisis
en Repositorios Multimedia

35

35

Contenidos

- Introducción
- Reconocimiento del locutor independiente del texto
- Reconocimiento del locutor dependiente del texto
- Reconocimiento del idioma



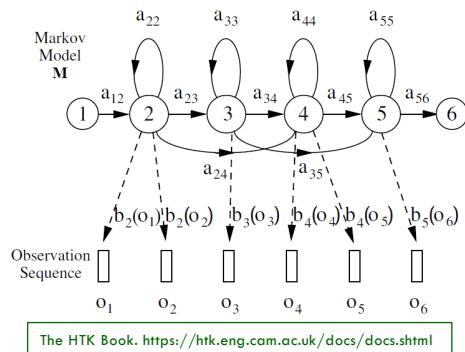
Reconocimiento de Locutor Dependiente de Texto

- El texto es conocido
 - ▣ Implica “colaboración” por parte del locutor
 - Para decir la frase que se supone que tiene que decir
- Aplicación típica: verificación de locutor en el acceso a recurso
 - ▣ Cuenta bancaria
 - ▣ Acceso a un recinto
 - ▣ ...
- Tipos de aplicación:
 - ▣ Texto fijo
 - Por ejemplo, un PIN, nombre y apellidos, etc.
 - ▣ Texto variable: *text-prompted systems*
 - Se presenta una frase por el sistema
 - Se pronuncia colaborativamente por el locutor
 - Ejemplo: diga “Mi voz es mi contraseña”



Reconocimiento de Locutor Dependiente de Texto: Estrategia

- La existencia de un texto conocido habilita el uso de HMMs
 - ▣ La secuencia de estados, en cierto modo, representa la secuencia de diferentes sonidos a través de la frase, **la cual es conocida**
 - ▣ El HMM introduce la evolución temporal de las características espectrales según va avanzando la frase



Reconocimiento de Locutor Dependiente de Texto: Estrategia

- El modelo HMM se entrena con locuciones del locutor, **conociendo el texto**



- ▣ La estructura de estados del HMM se elegirá dependiendo del texto que se sepa que se va a pronunciar
 - ▣ Modelos de palabra o frase completa (como en rec. dígitos ASR)
 - Se suele usar en sistemas de texto fijo
 - ▣ Modelos de fonemas concatenados
 - Se usa cuando se quiere generar una frase diferente cada vez en sistemas *text-prompted*, por ejemplo:
 - Intento de acceso 1: "Diga: mi voz es mi contraseña"
 - Intento de acceso 2: "Diga: accedo con mi propia voz"

Topologías

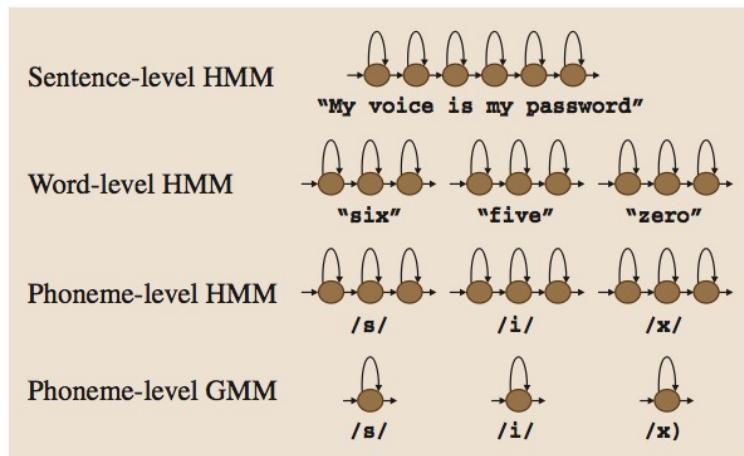


Fig. 37.1 Hidden Markov model (HMM) topologies

Reconocimiento de Locutor Dependiente de Texto: Estrategia

- Cálculo de puntuación (score) entre modelo y características
 - ▣ Idea: el modelo representa al locutor diciendo el texto
 - Si el texto no cambia entre el modelo (S) y la locución de test (C)...
 - La verosimilitud acústica con el HMM se deberá sobre todo al locutor
 - Porque el fichero de test y el modelo pronuncian el mismo texto
 - Si el locutor es el mismo, la verosimilitud será mayor
 - Si el locutor es diferente, la verosimilitud será menor



Efecto de la variabilidad Intersesión

- La variabilidad intersesión también afecta enormemente a los sistemas de reconocimiento de texto
 - Ejemplo: EER (cuanto más bajo mejor) dependiente del tipo de desajuste entre el fichero que entrenó el modelo y el fichero de test

Type of mismatch	Accuracy (EER) (%)
No mismatch	7.02
SNR mismatch	7.47
Channel mismatch	9.76
Lexical mismatch (LD2)	8.23
Lexical mismatch (LD4)	13.4
Complete lexical mismatch (LD6)	36.3

Table 37.1 Effect of different mismatch types on the EER for a text-dependent speaker verification task (after [37.4]). The corpus is from a pilot with 120 participants (gender balanced) using a variety of handsets. Signal-to-noise ratio (SNR) mismatch is calculated using the difference between the SNR during enrollment and testing (verification). For the purposes of this table, an absolute value of this difference of more than 10 db was considered mismatched. Channel mismatch is encountered when the enrollment and testing sessions are not on the same channel. Finally, lexical mismatch is introduced when the lexicon used during the testing session is different from the enrollment lexicon. In this case, the password phrase was always a three-digit string. LD0 stands for a lexical match such that the enrollment and testing were performed on the same digit string. In LD2, only two digits are common between the enrollment and testing; in LD4 there is only one common digit. For LD6 (complete lexical mismatch), the enrollment lexicon is disjoint from the testing lexicon. Note that, when considering a given type of mismatch, the conditions are matched for the other types. At EERs around 8%, the 90% confidence interval on the measures is 0.8%

Contenidos

- Introducción
- Reconocimiento del locutor independiente del texto
- Reconocimiento del locutor dependiente del texto
- Reconocimiento del idioma

Reconocimiento de idioma

- **Objetivo:** detección de idioma hablado en conversaciones ó grabaciones
- Aplicaciones:
 - ▣ Enrutamiento de llamadas (operadores, reconocedores específicos, etc.)
 - Ejemplo: dirigir una llamada a un operador en el idioma de la persona que habla en un *call center*
 - ▣ Clasificación de audios/llamadas
 - Ejemplo: indexación de contenidos en grandes repositorios de vídeos, para búsquedas o recomendación (publicidad) posteriores adaptadas a la región geográfica
 - ▣ Detección y monitorización de hablantes anónimos y/ó grupos de hablantes por el idioma hablado
 - Ejemplo: monitorización de idiomas en llamadas, típicamente para aplicaciones de inteligencia

Tipos de reconocedores de idioma

- Token-based (fonético/lingüístico)
 - ▣ PRLM (Phone Recognition followed by Language Modeling)
 - Varios PRLM en paralelo combinados: P-PRLM (Parallel PRLM)
- Spectral-based
 - ▣ Tecnologías muy parecidas a las de reconocimiento de locutor independiente de texto
 - Pero en lugar de entrenar modelos con locutores diferentes, se entrena con idiomas diferentes
 - Y se suelen usar características MFCC aumentadas (Shifted-Delta Cepstrum)
 - ▣ Tipos de modelos y sistemas:
 - GMM-UBM
 - i-vectors
 - Redes neuronales (x-Vectors)

Tokenizer: Phone Recognizer (PR)

- Convierte voz en secuencia acústica de fonemas reconocidos
 - SIN modelo de lenguaje (open loop / null grammar)
 - Aprovechar fonotáctica

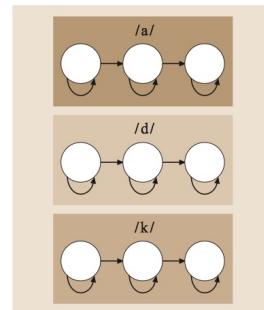


Fig. 41.6 HMM:
typical HMM
topology for
phonetic units

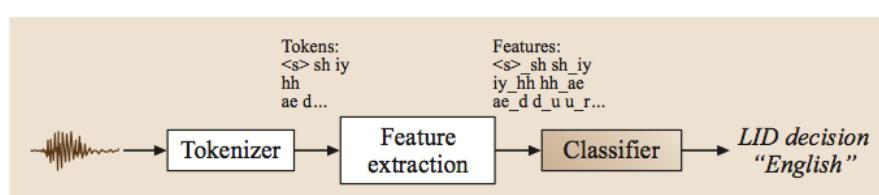


Fig. 41.4 Token-based language recognition

PRLM

- Aprende el modelo de idioma para un reconocedor fonético (*tokenizer*) dado
- La puntuación la da el modelo de lenguaje (*n*-gramas), que modela las características de las secuencias de fonemas dadas por el tokenizador (*phone recognizer*)

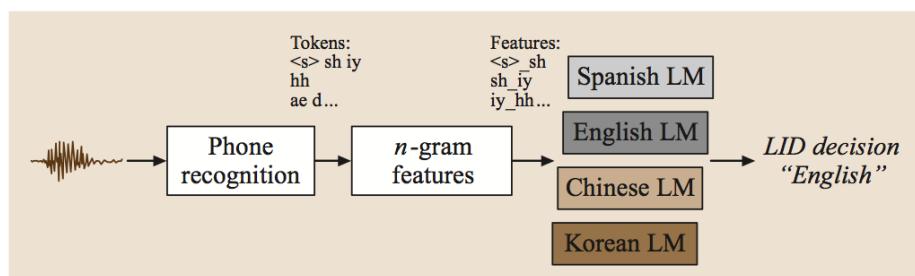


Fig. 41.5 PRLM: phone recognition followed by language modeling

PPRLM (Parallel PRLM)

- Varios tokenizadores en paralelo (explotan información complementaria debido a tokenizadores con colecciones de fonemas diferentes)
- Los modelos de lenguaje de los idiomas a reconocer calcular la verosimilitud (*likelihood*)

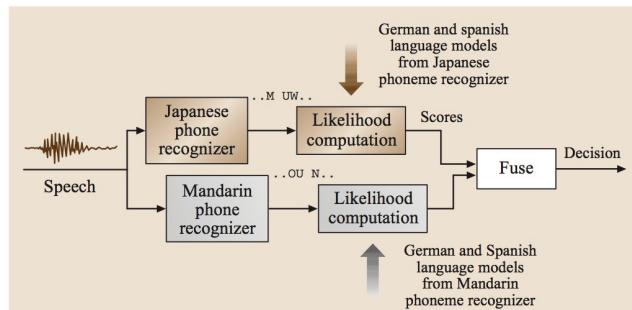
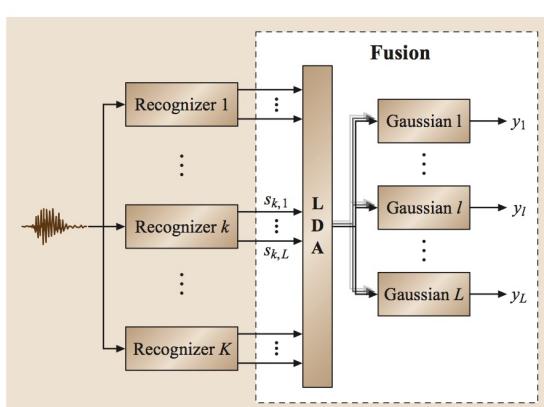


Fig. 41.8 PPRLM: parallel phone recognition followed by language modeling

Fusión de sistemas



The scores of the core language recognizers were stacked into a 286-dimension feature vector and fed to the fusion back-end for dimension reduction (linear discriminant analysis) and classification (seven Gaussians with pooled diagonal covariances). The outputs of the back-end were treated as estimated target language likelihoods and log likelihood ratios were formed by taking the log of the quantity in (41.28). These scores were treated as the final outputs of the system.

Fig. 41.9 Gaussian-based fusion for language recognition

Resultados NIST LRE 2005

- NIST Language Recognition Evaluation
 - Benchmark “estándar” para comparar sistemas de reconocimiento de idioma
 - Gram complejidad, múltiples participantes de todo el mundo

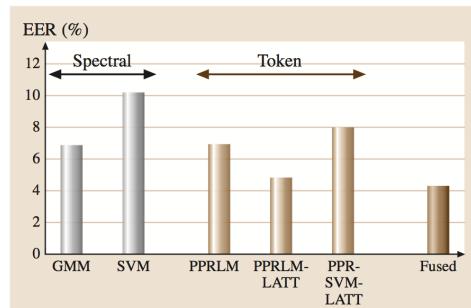


Fig. 41.11 Performance (percentage EER) of spectral, token, and fused language recognition systems on the 2005 NIST LRE primary condition test

Tecnologías de Reconocimiento de Idioma

- Sistemas PPRLM (2000-2008)
 - Múltiples reconocedores fonéticos y de lenguaje en paralelo
 - Tremendamente costosos (cómputo)
- Sistemas i-vector (2006-2014)
 - Sistemas acústicos basados en envolvente espectral, extremadamente ligeros y eficientes
- Deep Neural Networks (DNNs) (2012-)
 - Múltiples capas ocultas
 - Uso como extractores de *embeddings* (*x*-Vectors)
 - Pero también como clasificadores de idioma
 - Redes recurrentes (LSTM)
 - Gran rendimiento también en duraciones cortas (~3s)

Reconocimiento de Idioma con LSTMs



RESEARCH ARTICLE

Language Identification in Short Utterances Using Long Short-Term Memory (LSTM) Recurrent Neural Networks

Ruben Zazo*, Alicia Lozano-Díez, Javier Gonzalez-Dominguez, Doroteo T. Toledano, Joaquín González-Rodríguez

ATVS-Biometric Recognition Group, Universidad Autónoma de Madrid, Madrid, Spain

* ruben.zazo@uam.es



Abstract

Long Short Term Memory (LSTM) Recurrent Neural Networks (RNNs) have recently outperformed other state-of-the-art approaches, such as i-vector and Deep Neural Networks (DNNs), in automatic Language Identification (LID), particularly when dealing with very short utterances (~3s). In this contribution we present an open-source, end-to-end, LSTM RNN system running on limited computational resources (a single GPU) that outperforms a reference i-vector system on a subset of the NIST Language Recognition Evaluation (8 target languages, 3s task) by up to a 26%. This result is in line with previously published research using proprietary LSTM implementations and huge computational resources, which made these former results hardly reproducible. Further, we extend those previous experiments modeling unseen languages (out of set, OOS, modeling), which is crucial in real applications. Results show that a LSTM RNN with OOS modeling is able to detect these languages and generalizes robustly to unseen OOS languages. Finally, we also analyze the effect of even more limited test data (from 2.25s to 0.1s) proving that with as little as 0.5s an accuracy of over 50% can be achieved.



Máster en Big Data y Data Science

Indexación, Búsqueda y Análisis
en Repositorios Multimedia

52

52



Máster en Big Data y Data Science

Indexación, búsqueda y análisis
en repositorios multimedia

Indexación de audio: Reconocimiento de Locutor e Idioma

< audias >

Audio, Data Intelligence and Speech
<http://audias.ii.uam.es>

Daniel Ramos Castro

Contrib. de Doroteo Torre, Joaquín González, Alicia Lozano

53