University of The Basque Country

Universidad del País Vasco
Euskal Herriko Unibertsitatea

Doctoral Thesis

Frame-Level Features Conveying Phonetic Information for Language and Speaker Recognition

Author:
Mireia Diez Sánchez

Supervisors:
Dra. Amparo Varona
Dr. German Bordel

A thesis submitted in fulfilment of the requirements for the degree of Doctor

in the

Software Technologies Working Group
Department of Electricity and Electronics

June 2015

UNIVERSITY OF THE BASQUE COUNTRY

Abstract

Faculty of Science and Technology
Department of Electricity and Electronics

Doctoral Thesis

Frame-Level Features Conveying Phonetic Information for Language and Speaker Recognition

by Mireia Diez Sánchez

This Thesis, developed in the Software Technologies Working Group of the Department of Electricity and Electronics of the University of the Basque Country, focuses on the research field of spoken language and speaker recognition technologies.

More specifically, the research carried out studies the design of a set of features conveying spectral acoustic and phonotactic information, searches for the optimal feature extraction parameters, and analyses the integration and usage of the features in language recognition systems, and the complementarity of these approaches with regard to state-of-the-art systems. The study reveals that systems trained on the proposed set of features, denoted as Phone Log-Likelihood Ratios (PLLRs), are highly competitive, outperforming in several benchmarks other state-of-the-art systems. Moreover, PLLR-based systems also provide complementary information with regard to other phonotactic and acoustic approaches, which makes them suitable in fusions to improve the overall performance of spoken language recognition systems.

The usage of this features is also studied in speaker recognition tasks. In this context, the results attained by the approaches based on PLLR features are not as remarkable as the ones of systems based on standard acoustic features, but they still provide complementary information that can be used to enhance the overall performance of the speaker recognition systems.

UNIVERSIDAD DEL PAÍS VASCO

Resumen

Facultad de Ciencia y Tecnología
Departamento de Electricidad y Electrónica

Tesis doctoral

Características a Nivel de Trama con Información Fonética para el Reconocimiento de la Lengua y del Locutor

por Mireia Diez Sánchez

Esta tesis, desarrollada en el Grupo de Trabajo en Tecnologías del Software del Departamento de Electricidad y Electrónica de la universidad del País Vasco, se centra en el campo de investigación del reconocimiento de la lengua y del locutor.

Más específicamente, la investigación llevada a cabo estudia el diseño de un conjunto de características que aportan información espectro-acústica y fonotáctica, busca la configuración óptima para la extracción de las mismas, y analiza la integración y uso de las características en sistemas de reconocimiento de la lengua, así como la complementariedad de estas aproximaciones con respecto a otros sistemas acústicos y fonotácticos actualmente punteros. El estudio revela que los sistemas entrenados con estas características, que podríamos denominar cocientes de log-probabilidades de fonemas (PLLRs, de sus siglas en inglés), son altamente competitivos, superando en rendimiento a otros sistemas. Además, los sistemas basados en estas características también proporcionan información complementaria con respecto a otras aproximaciones fonotácticas y acústicas, lo que los hace adecuados en fusiones, para mejorar el rendimiento general de los sistemas de reconocimiento.

También se estudia el uso de las características en tareas de verificación de locutor. En este contexto, los resultados obtenidos por las aproximaciones basadas en PLLRs no son tan notables como los obtenidos con características acústicas, pero proporcionan información complementaria que puede ser utilizada para mejorar el rendimiento general de los sistemas de reconocimiento de locutor.

EUSKAL HERRIKO UNIBERTSITATEA

Laburpena

Zientzia eta Teknologia Fakultatea
Elektrizitate eta Elektronika Saila

Leiho-Mailako Karakteristikak Informazio Fonetikoarekin Lengoai eta Esatarien Egiaztapenerako

Tesi Doktorala

Mireia Diez Sánchez

Euskal Herriko Unibertsitateko Elektrizitate eta Elektronika Saileko Software Teknologien Lan Taldean garatutako tesi hau, hizkuntza eta hiztun ezagutze arloen ikerketan oinarritzen da.

Zehazki, ikerketak, informazio akustiko-espektrala eta fonotaktikoa daramaten ezaugarri multzo bat aztertzen du, erauzketa konfigurazio hoberena bilatuz, egungo hizkuntza-ezagutze sistemetan integratzeko aukerak aztertuz eta beste sistema akustiko eta fonotaktikoekin duen osagarritasuna aztertuz. Ikerketak erakusten du fonemen log-probabilitate erlazio (ingelesetik, PLLRak) izendatutako ezaugarri hauekin entrenatutako sistemak, datu-base ugarietan, oso emaitza onak lortzen dituztela, gaur eguneko beste sistema askorekin konparatuz. Gainera, ezaugarri hauetan oinarritutako sistemak beste hurbilketa akustiko eta fonotaktikoekiko informazio osagarria eskaintzen dutenez, sistema orokorren errendimendua hobetzeko egokiak dira.

Ezaugarri hauen erabilpena hiztun-ezagutze sistemetan ere ikertzen da. Kontextu honetan, PLLRak erabiltzen dituzten sistemen emaitzak ez dira sistema akustikoak lotzen dituztenak bezain ikusgarriak, baina hurbilpenak oraindik informazio osagarria eskaintzen du, sistema orokorren ekimena hobetzeko erabilgarria izan litekeena ere.

Acknowledgements

It's been more than five years since I started this journey, five not always easy but still incredible years, in which I've grown, I've changed and most of all I've learned. I owe that to a lot of people, I've been lucky enough to find a lot of people to learn from along the way.

First of all thanks to the Department of Education, Universities and Research of the Basque Government for the pre-doctoral fellowship (BFI09.263/AE) that allowed me start my research career. Thanks to the Basque Government too for the economic support conceded by the SAIOTEK projects S-PE11UN065, S-PE12UN055 and S-PE13UN105. To the University of the Basque Country for the support provided by the group grants GIU10/18 and GIU13/28. And to the Spanish Ministry of Science and Innovation (MICINN) for the project TIN2009-07446.

Thanks to all GTTS members. Thanks for putting trust in me and introducing me to the *speech technologies world*. Thanks for making me feel part of the group since day one, for all the ideas, meetings and debates that have made sense of all these years of work. Germán, for your refreshing new points of view when I was in need of them, and for shaping this manuscript. Luisja, for that first e-mail that sparked my interest for this field and for the re-revisions, I don't think I would have learned to write anything decent if it wasn't for you. Mikel, zure patzientziagatik, kodigoa eta artikuloak destripatzen egon garen hainbeste orduengatik, zure dedikazio eta entusiasmoa transmititzeagatik, dakidan gehiena zuri esker da. Amparo, for being much more than a thesis advisor, for the orientation, the organization, for all the chatting hours, and the big support you've been through all these years. Guztioi, benetan, mila esker.

I had the amazing opportunity of working with people from other research groups from all around the world. Thanks to the members of ViVoLab for letting me spend some time with the group, if only for a short time, when I was just starting. Carlos Vaquero, thanks for opening up doors for me. Thanks to the BOSARIS II team, for the gained experience and the bunch of experiences. And most of all, thanks to the BUT Speech@FIT group. Honza Černocký, thanks for giving me the chance of joining the team for an internship, I cannot think of a better working environment. Thanks to the whole group, and especially to Jan, Oldřich & Lukáš, for being so

welcoming, for the patience with all the questions, for the guidance, the lessons and for all the good times, you guys rock.

I cannot forget about my everyday mates, the ones that make you come to work with a smile even when you can foresee a "12 hours in the lab" day. To all the ones with whom I've shared lab or corridor, some joy, and probably some frustration too; particularly to Silvia, Héctor, Olaso and Carlos: thanks for all those coffees, and all the waves of madness. Inari, neither the university, nor these years would have been the same without you, thanks for being always there for me.

To all those friends who have supported me from "the other side" for understanding my absences, supporting me during the bad times and celebrating the good ones. Larraitz, Sandra and Jona, for making me feel you all near me even when I was far away. Mertxe, por tu amistad. Irati, a big and simple thanks for being you. Gorka, for everything along the path we walked side by side. Amaiur, for the great support you've been so many times. Nano, thanks for all the advice and for showing me how to look at life and the thesis differently. And, of course, to my two biggest treasures, Udane and Alex, for giving me strength and making me smile.

Por último: aita y ama, gracias por todo, porque sin la educación que escogisteis para mí, los valores que me habéis inculcado y el apoyo incondicional que siempre me habéis dado, no hubiera llegado hasta aquí, ni sería la persona que soy hoy en día.

And, as it wouldn't be me without a touch of the bizarre, thanks to all kinds of music, desserts and chocolate, because without them I would have probably lost the little sanity that I've got left :).

"Only as high as I reach can I grow,
only as far as I seek can I go,
only as deep as I look can I see,
only as much as I dream can I be"

- Karen Ravn

Contents

List of Figures

List of Tables

Abbreviations

TECHINICAL TERMS

ActDCF	**Act**ual **D**etection **C**ost **F**unction
$\mathbf{C}_{\text{avg}}$	**Av**era**g**e **C**ost
$\mathbf{C}_{\text{LLR}}$	**L**og-**L**ikelihood **R**atio **C**ost
CMS	**C**epstral **M**ean **S**ubtraction
CTS	**C**onversational **T**elephone **S**peech
DCF	**D**etection **C**ost **F**unction
DCT	**D**iscrete **C**osine **T**ransform
DET	**D**etection **E**rror **T**radeoff
DNN	**D**eep **N**eural **N**etworks
DFT	**D**iscrete **F**ourier **T**ransform
EER	**E**qual **Error R**ate
EM	**E**xpectation **M**aximization
$\mathbf{F}_{\text{act}}$	Acutal Relative Confusion
FFT	**F**ast **F**ourier **T**ransform
GMM	**G**aussian **M**ixture **M**odel
HMM	**H**idden **M**arcov **M**odel
HTK	**H**idden Markov Model **T**ool**K**it
IPA	**I**nternational **P**honetic **A**lphabet
JFA	**J**oint **F**actor **A**nalysis
LE-GMM	**L**inearized **E**igenchannel **GMM**
LID	**L**anguage **ID**entification
LLR	**L**og-**L**ikelihood **R**atio
LR	**L**ogistic **R**egression
LRE	**L**anguage **R**ecognition **E**valuation
MAP	**M**aximum **A** **P**osteriori
MD	**M**inumum **D**ivergence
MinDCF	**Min**imum **D**etection **C**ost **F**unction
ML	**M**aximum **L**ikelihood
MFCC	**M**el **F**requency **C**epstral **C**oefficients
MMI	**M**aximum **M**utual **I**nformation
NN	**N**eural **N**etwork
OOS	**O**ut **O**f **S**et
PCA	**P**rincipal **C**omponent **A**nalysis
PLLR	**P**hone **L**og-**L**ikelihood **R**atios
PLDA	**P**robabilistic **L**inear **D**iscriminant **A**nalysis
PLP	**P**erceptual **L**inear **P**rediction
PPRLM	**P**arallel **P**hone **R**ecognition **L**anguage **M**odeling
PRLM	**P**hone **R**ecognition **L**anguage **M**odeling
RASTA	**R**el**A**tive **S**pec**T**r**A**
RATS	**R**obust **A**utomatic **T**ranscription of **S**peech
SD	**S**hifted **D**elta
SDC	**S**hifted **D**elta **C**epstrum
SLR	**S**poken **L**anguage **R**ecognition
SNR	**S**ignal **N**oise **R**atio
SR	**S**peaker **R**ecognition
SRE	**S**peaker **R**ecognition **E**valuation
SVM	**S**upport **V**ector **M**achine
TRAP	**T**emporal **P**atterns
UBM	**U**niversal **B**ackground **M**odel
VAD	**V**oice **A**ctivity **D**etector
VOA	**V**oice **O**f **A**merica

INSTITUTIONS

BUT	**B**rno **U**niversity of **T**echnology
EHU	**E**uskal **H**erriko **U**nivertsitatea *University of the Basque Country*
GTTS	**G**rupo de **T**rabajo en **T**ecnologías **S**oftware *Software Techonologies Working Group*
ISCA	**I**nternational **S**peech **C**ommunication **A**ssociation
JHU	**J**ohn **H**opkins **U**nivertisy
L2F	**L**aboratorio de sistemas de **L**íngua **F**alada *Laboratory of Spoken Language systems*
LDC	**L**inguistic **D**ata **C**onsortium
NIST	**N**ational **I**nstitute of **S**tandard **T**echnologies
RTTH	**R**ed **T**emática en **T**ecnologías del **H**abla *Spanish Thematic Network on Speech Technology*
SIG_IL	**S**pecial **I**nterest **G**roup on **I**berian **L**anguages

A los que entienden ablakistriki
y chuchurriña.

Sinopsis

El reconocimiento de la lengua y el reconocimiento de locutor son tareas de reconocimiento de patrones que consisten en determinar, mediante métodos computacionales, la lengua hablada y la identidad del hablante en una señal de voz, respectivamente.

Los sistemas de reconocimiento tienen una estructura general que consta de diferentes módulos. En primer lugar se encuentra el módulo de extracción de características, que toma como entrada una señal de audio y obtiene un conjunto de características que recogen información, entre otras cosas, sobre la lengua y sobre el hablante. A continuación se encuentra el clasificador. Entrenado durante la fase de modelado, este módulo toma como entrada las características de cada señal de voz y obtiene una serie de puntuaciones, indicando su similitud con cada modelo objetivo. Las puntuaciones deben ser ajustadas en el módulo de calibrado para ecualizarlas de tal modo que puedan enfrentarse a un mismo umbral sobre el que tomar las decisiones finales.

La mayor dificultad de las tareas de reconocimiento de la lengua y del locutor reside en extraer de la señal de voz información relevante para el reconocimiento, descartando la relativa a otras fuentes de variabilidad, reunidas generalmente bajo la denominación de *variabilidades de canal*. Estos cambios o variaciones de la señal pueden ser originados por el aparato de grabación o el canal de transmisión, ruido ambiente, estados de ánimo o condiciones físicas del hablante, variabilidad relacionada con el locutor en el caso de la lengua, o de la lengua en el caso del locutor, etc.

Los estudios realizados en esta tesis se han enfocado principalmente a la tarea de reconocimiento de la lengua y han sido extendidos posteriormente al reconocimiento del locutor.

Las diversas aproximaciones desarrolladas para el reconocimiento de la lengua, se basan en diferentes tipos de características. En términos generales, estas características abarcan propiedades espectrales, fonéticas, acústicas, temporales o fonotácticas de la señal, prosodia u otras características de índole más compleja como la construcción de palabras y frases. La mayoría de los sistemas hacen uso

de dos conjuntos principales de características: acústico-espectrales y fonéticas. Las características acústico-espectrales (también denominadas de *bajo nivel*) extraen la información analizando el espectro de la señal mediante ventanas de unos 20-30 milisegundos, intentando reproducir los mecanismos de la percepción humana de los sonidos. Las características fonéticas (o de *alto nivel*) extraen información relativa a los fonemas contenidos en la señal. Dado que cada lengua tiene su propio inventario fonético, es fácil intuir cómo la identificación de los fonemas puede ayudar en las tareas de reconocimiento. En muchos casos los sistemas finales de reconocimiento de la lengua son combinaciones de diversos sistemas (basados en ambos tipos de características) fusionados en la parte final del proceso, conocida como calibrado, en la que se transforman y combinan las puntuaciones.

Esta tesis se centra en la extracción de unas características con información fonética a nivel de trama, con el objetivo de combinar aportes de ambos tipos de aproximaciones en las propias características. El trabajo desarrollado abarca la optimización e integración de las características en sistemas de reconocimiento de la lengua y del locutor.

Las características propuestas, denominadas "relaciones de log probabilidades de fonemas" (PLLRs, de sus siglas en inglés), son obtenidas a partir de probabilidades fonéticas. Dada una señal de audio y un decodificador fonético con un inventario de N fonemas, se supone que el decodificador proporciona la probabilidad acústica p para cada una de las unidades fonéticas i ($1 \leq i \leq N$), en cada ventana (trama) correspondiente al instante de tiempo t, $p(i|t)$. El vector N-dimensional obtenido es aquel que, según los parámetros del decodificador, mejor describe el contenido espectral de la ventana de análisis. Geométricamente, este vector puede verse como un punto dentro del simplex N-1 estándar, donde cada vértice representa unidades fonéticas puras y los bordes mezclas de las unidades fonéticas que unen.

Los PLLRs se calculan para cada fonema i y para cada ventana en el instante de tiempo t, según la siguiente expresión:

$$PLLR(i|t) = \log \frac{p(i|t)}{\frac{1}{(N-1)}(1 - p(i|t))} \qquad i = 1, ..., N \tag{1}$$

Los PLLRs, al ser características obtenidas a nivel de trama, son fácilmente integrables en sistemas del estado del arte basados en otro tipo de características

espectrales. En nuestros estudios se han utilizado bajo la aproximación conocida como Total Variability Factor Analysis o *i-vector*, una de las técnicas que mejores resultados proporcionan, y por lo tanto una de las técnicas más utilizadas.

Los primeros análisis se centran en la optimización de los parámetros de extracción de las características PLLR. Se explora el uso de formulaciones próximas a la definición presentada, así como la integración de información con un mayor contexto temporal alrededor de la ventana, al igual que se hace con otras características espectrales como los Mel Frequency Cepstral Coefficients (MFCC). Por otro lado, se estudia la aplicación de diferentes técnicas de compensación de variabilidad a nivel de extracción de características, y el uso de diversos decodificadores fonéticos para el cómputo de los PLLRs.

Una vez encontrada la configuración óptima, haciendo uso de coeficientes dinámicos de primer orden, los resultados obtenidos por los sistemas basados en las características PLLR son contrastados con aquellos obtenidos por sistemas acústicos basados en MFCCs, que aplican el mismo modelado, y con sistemas fonotácticos, que utilizan los mismos decodificadores como base para la extracción de sus características. Se analizan los resultados en cuatro bases de datos diferentes: las tres últimas bases de datos proporcionadas por el National Institute of Standards Technology (NIST), para las NIST 2007, 2009 y 2011 Language Recognition Evaluations (LRE), de gran relevancia en el campo de investigación de reconocimiento de la lengua; y la base de datos proporcionada por GTTS para Albayzin 2010 LRE, en la que se analizan los resultados en entornos y tipos de señal de diferente naturaleza. Los resultados obtenidos son consistentes en todas las bases de datos. Los sistemas basados en PLLRs obtienen en la mayoría de los casos tasas de error inferiores a las de los sistemas del estado del arte, lo que pone de manifiesto la utilidad de estas características.

Por otro lado, también se procede a la fusión de sistemas. Se fusionan las puntuaciones del sistema basado en PLLRs con las del sistema acústico, revelando una alta complementariedad entre ambos sistemas. La fusión del sistema basado en características PLLR con sistemas fonotácticos evidencia también una alta complementariedad. Por último, la fusión de los tres sistemas proporciona mejoras con respecto a cualquier fusión de sistemas por pares. Este resultado es muy significativo, ya que muestra que la información aportada por los PLLRs puede utilizarse

para mejorar el rendimiento total de sistemas del estado del arte, cualesquiera que sean las características utilizadas.

El siguiente estudio se centra en la reducción de dimensionalidad de los PLLRs, debido a que ésta es relativamente grande en comparación con otros vectores de características espectrales. La reducción podría facilitar la aplicación de otras técnicas en la extracción de PLLRs. Se analizan varias aproximaciones supervisadas, basadas en el conocimiento de la naturaleza fonética de la lengua usada en el decodificador. Por otro lado, se estudian también técnicas no supervisadas, que reducen el tamaño del vector basándose en correlaciones entre fonemas o en la frecuencia de fonemas, así como la técnica conocida como Principal Component Analysis (PCA), que realiza una transformación ortogonal en el espacio de los PLLRs. Se concluye que el vector de características puede ser reducido a prácticamente un tercio de su dimensionalidad (alcanzando una dimensionalidad comparable con otras aproximaciones espectrales) mediante técnicas supervisadas, sin degradar significativamente el rendimiento del sistema. Por otro lado, la aplicación de PCA revela que el vector puede ser reducido mejorando además los resultados obtenidos por el sistema.

Una vez encontrada la forma óptima de reducir el vector de características, se procede a estudiar la mínima dimensionalidad que se puede alcanzar mediante PCA (manteniendo un compromiso con los resultados obtenidos) con el objetivo de calcular secuencias de derivadas (Shifted Delta, en inglés) sobre los PLLRs, ya que su uso es común sobre características espectrales (MFCCs) en sistemas de reconocimiento de la lengua. Definida la mínima dimensionalidad, se procede al barrido de parámetros de extracción de las secuencias de derivadas. Finalmente, la aplicación de la combinación óptima revela que su uso es acertado también sobre los PLLRs, ya que la información que aportan mejora significativamente los resultados obtenidos por el sistema.

El descubrimiento de que la reducción mediante PCA proporciona mejoras en el sistema, ha derivado posteriormente en el análisis del espacio de los PLLRs, con el fin de entender posibles orígenes de este resultado. El estudio revela que los PLLRs están confinados en un sub-espacio limitado por un hiperplano que es asintóticamente perpendicular a los ejes de los PLLRs. Este efecto limitaría presuntamente el *movimiento* de los PLLRs respecto a los ejes y por tanto, la información que proporcionan. Se diseña un método de proyección que elimine el efecto. La proyección

de los PLLRs potencia el rendimiento del sistema de forma significativa, resultados que son mejorados aún más si la proyección se combina con la transformación PCA, debido a la decorrelación de parámetros que realiza esta última, que optimiza la distribución de las características para el tipo de modelado utilizado.

Posteriormente, fruto de una estancia con el grupo Speech@FIT de la universidad de Brno, se analiza el rendimiento del sistema basado en PLLRs proyectados, bajo aproximaciones basadas también en i-vectors, pero con modelado final utilizando regresión logística y redes neuronales, y sobre la base de datos Robust Automatic Transcription of Speech (RATS). Los resultados obtenidos con señales ruidosas y de duraciones más cortas revelan, una vez más, la alta competitividad de los sistemas basados en PLLRs.

A continuación, se procede a combinar la proyección de las características con la aplicación de las secuencias de derivadas. Los resultados obtenidos en las bases de datos NIST 2007 y 2011 LRE muestran que la combinación de ambas técnicas es acertada.

Finalmente, se estudia el uso de los PLLRs en sistemas de reconocimiento de locutor. Se procede, en primer lugar, a la optimización de parámetros de extracción de PLLRs para estos sistemas. Una vez encontrada la combinación óptima, haciendo uso también de coeficientes dinámicos de primer orden, se comprueban los resultados de los sistemas utilizando i-vectors junto con *Probabilistic Linear Discriminant Analysis* (PLDA). Para ello se emplean las dos últimas bases de datos proporcionadas por el NIST para las NIST 2010 y 2012 (Speaker Recognition Evaluation) SRE. En este caso, se observa que los sistemas basados en PLLRs no obtienen resultados tan relevantes como los obtenidos por sistemas basados en características acústico-espectrales como los MFCCs. Aun así, al proceder a la fusión de sistemas, se obtiene una ganancia en varias condiciones de evaluación, lo que revela que la complementariedad de las características aporta información útil para mejorar el rendimiento del sistema.

Chapter 1

Introduction

1.1 Context

Spoken language verification and speaker verification are pattern recognition tasks that consist of recognizing, by computational means, the language spoken and the speaker speaking in an utterance, respectively.

It should be noted that recognition, identification and verification are different tasks. In identification, it is assumed that the utterance corresponds to one of a certain set of N models, and the task consists on selecting one of the N models as the true identity. A verification system, instead, must decide whether a certain utterance contains speech of a target model or not. That is, each utterance is tested against each model independently. Both tasks are recognition tasks. Even if this work focuses on verification tasks, the term recognition will be used in the manuscript, as its use is a common practice in the field.

A distinction should be made also between closed and open verification tasks. In the case of a closed task, all the target models considered in the set are known, whereas in open set tasks, "none of the target models", that is, a model considering *unknown languages* or *unknown* speakers is also considered in the set of models.

The main complexity of language and speaker recognition tasks comes from dealing with undesired variabilities present in the utterances due to several factors: the recording device, the transmission channel, environmental noises, mood or physical conditions, differences among speakers in the case of language verification, different

languages or aging effects in the case of speaker verification, etc. All these, commonly summarized as channel variabilities, pose a strong difficulty for all the tasks related to speech recognition, like Spoken Language Recognition (SLR) and Speaker Recognition (SR). Extracting informative features robust against those variabilities or designing modeling techniques capable of pattern after the desired variabilities, while discarding the noisy ones, are the main focus of research in both tasks.

1.1.1 Structure of Recognition Systems

Recognition systems have a common basic structure, that can be outlined as follows. First of all, it is necessary to record and organize a set of speech signals (or to reuse an existing one) on top of which the system is going to be built. Data should be divided into several sets, one for each of the main stages of SR/SLR system building (see Figure 1.1): *the training* of the different components present in the recognition system, *the development* and tuning of the parameters of the different modules, and *the evaluation*, where system performance is measured on a test benchmark (an independent set of speech signals). Therefore, in general terms, training, development and evaluation data are necessary. The assembling of the sets is a crucial step. Unbalanced sets could pose several problems, like the over-training of the system for certain types of variabilities (under-training for others) causing biased performances and non-robust systems. In optimal circumstances, all the subsets would consist of balanced data representative of the overall variability present in the application scenario. Once the data is available, verification systems can be built.

The structure of the SLR/SR systems comprises four main modules (see Figure 1.1):

- Feature/token extractor. The first module of verification systems takes as input the utterance (speech) and aims to concentrate in few and, as far as possible, independent (that is, uncorrelated) parameters/features the information relevant to the classification task.
- Classifier. Built during the training stage, takes the features as input and scores feature/token sequences with regard to the target models.
- Backend transformation. This module is assembled on the development stage. It performs a transformation on the raw scores to normalize/calibrate them, that allows us to use a single threshold for all the targets and makes the system work at the desired application point.
- Decision maker. Provides a final decision (hard decision: accept/reject) for each target model on top of the transformed scores.

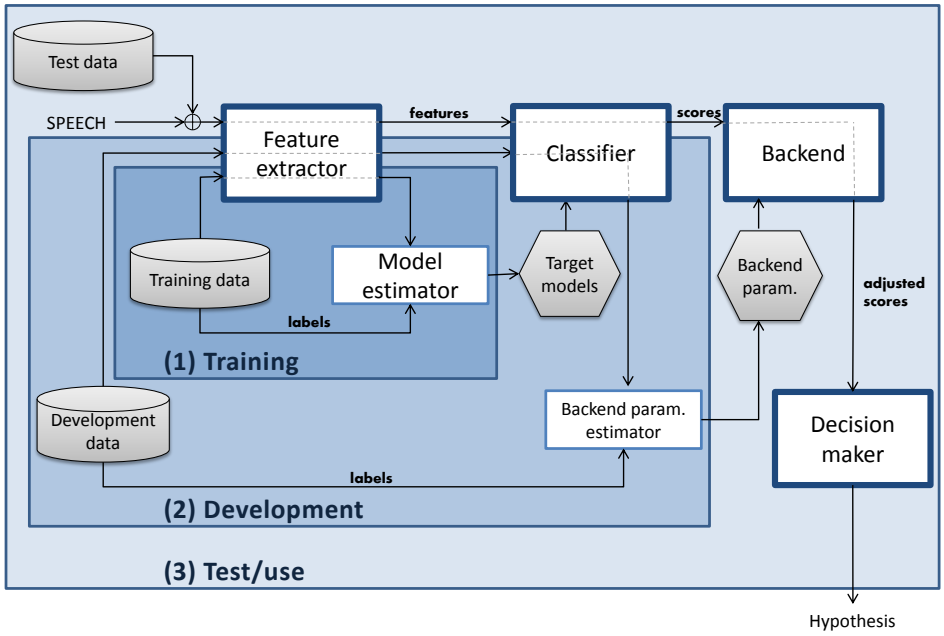


FIGURE 1.1: Structure of a recognition system.

In real systems, the distinction between the different modules is not always easy to define, e.g. some classifiers applied on top of features can be considered as higher level feature extractors, which are then used to fed a final classifier. Also, some classifiers provide normalized scores, suppressing the need of an extra backend parameter estimation stage.

1.1.2 Evolution

All the technologies in the area of language and speaker verification have evolved significantly in the last years. These progresses are due to different factors, that are briefly described in the following paragraphs.

On the one hand, the great effort in database creation/enlargement has promoted a richer benchmark for research, studies and system development. In this regard, it is remarkable the contribution of the National Institute of Standards Technology (NIST) by organizing a series of Language and Speaker Recognition Evaluations (LRE and SRE, respectively). These evaluations, starting in 1996 in the case of LRE and in 1997 for SRE, and held on a regular basis, have been —and still are— outstanding benchmarks for the research community. The contribution of NIST

is not limited to providing the databases, as the evaluations also evolve, posing challenging tasks that encourage groups from the research community to continuously improve their systems. The LRE datasets consist of spontaneous conversations collected through telephone (narrow-band) channels involving two speakers. These benchmarks have consistently grown from evaluation to evaluation in terms of amount of data, and have evolved regarding the target languages, ranging from nine up to twenty four. Evaluation tracks involved signals of 3, 10 and 30 second nominal durations. Regarding SRE, NIST databases started providing telephone conversational excerpts recorded over telephone channels, and increasingly added more kinds of recordings, including (in the last SREs) telephone data recorded over microphone channels and conversational speech data from interviews recorded over far-field microphones. The amount of trials has also been constantly increasing on each new release, reaching volumes that comprise recordings of thousands of speakers and millions of trials. As for language recognition, test segments are also divided into three subsets, according to their nominal duration.

Other resources have also promoted language and speaker recognition research. The KALAKA database was built for the Albayzin 2008 Language Recognition Evaluation[1], following the general approach of NIST LRE benchmarks, with some distinctions: while NIST LRE signals consisted of spontaneous conversations collected over telephone (narrow-band) channels, KALAKA datasets consisted of signals extracted from (wide-band) TV shows, including both planned and spontaneous speech in diverse environment conditions involving a varying number of speakers. The target language set focused on the 4 official languages spoken in Spain. The KALAKA2 and KALAKA3 databases, constructed for Albayzin 2010 and 2012 LREs, modified the scope of the evaluations. In the case of Albayzin 2010, an evaluation track was included to evaluate recognition of signals recorded in noisy environments, and included two more target languages. Albayzin 2012 pursued more ambitious challenges, as KALAKA3 consisted of audio data extracted from YouTube videos, and added evaluation tracks containing target languages for which no training materials were provided, increasing the amount of target languages up to ten.

The RATS dataset was designed to perform LRE and SRE in challenging scenarios, focusing on noisy environments. RATS provides data retransmitted through 8 different communication channels covering target languages. The utterances consist of selected signals from Callfriend and Fisher collections, previous NIST datasets and new conversational telephone speech.

On the other hand, the increase in computational power has enabled the usage of more complex algorithms for feature extraction and system modeling e.g. the use of

[1]The Albayzin evaluations were framed in the "Jornadas en Tecnologías del Habla" organized by the Spanish Thematic Network on Speech Technology (RTTH), and the ISCA Special Interest Group on Iberian Languages (SIG-IL)

supervectors as features or the training of variability matrices in high dimensional spaces. Also, progress has been made from using a single phone decoder to obtain 1-best decoding based approaches, to using lattices in n-grams of different orders extracted from multiple phone recognizers in parallel, to be used as features in further complex modeling techniques.

Nowadays, Deep Neural Networks (DNNs) are gaining strength in all the stages of SLR and SR systems, from feature extraction to variability compensation, modeling, calibration or even as multiple stages at once. The high computational power attained by current number crunching technology and the big amount of available resources seem to have reached the level needed to train complex and powerful DNNs, which makes them likely to be the path to follow in future research.

1.1.3 Applications

Progress in the field has allowed the development of a wide range of systems and applications for both spoken language and speaker recognition, which are becoming more and more common in the technology surrounding us. The possible applications include, among others:

- Phone service automation
- Phone call filtering
- Search and indexing of audiovisual resources
- Product customization
- Intelligence
- Forensics
- Speech preprocessing in multilingual dialog systems
- Speech preprocessing for suitable model/dictionary selection in data recovery systems
- Security

1.2 Motivation and Objectives of the Work

This thesis started like any other journey, with a destination and motivation to get there, a vision of the beginning of the path that should be taken, but only a slight idea of the forthcoming route and the things that would be found along the way. That destination, our former objective, was to build a competitive language recognition system.

The first thing to do before traveling, before even being able to write a plan, is to document about the things you can do, and the experiences others have had while going there. After adequate documentation, the "must sees" clearly showed up. I first toke part in developing a state-of-the-art language recognition system using NIST LRE benchmarks [44]. This first work, that took me into the *speech technologies world*, was framed in the Final Year Project and consisted on building a GMM-MAP language verification system, using NIST 2007 LRE as benchmark.

The closeness of SLR and SR tasks pushed me into the speaker recognition field, leading also to the construction of speaker recognition systems using NIST 2008 SRE [45], developed on the Master Thesis, which gave me some knowledge of the sometimes subtle, other times significant differences between both tasks.

In the meantime, speaker diarization showed up. The application, complexity, combination of several steps to build a system and the bunch of techniques caught our attention. We spent almost a year struggling with tons of papers, software and different ideas, that led to the construction of a dot-scoring based system, that was presented to the Albayzin 2010 speaker diarization evaluation [47, 48]. Even though I remember that time as a truly exciting period, you cannot spread yourself too thin, so the field was finally left aside to focus on language recognition again.

Albayzin evaluations provided an interesting benchmark to follow the research in the language recognition field [128, 133–135, 152]. Related to Albayzin evaluations, we also shared the experience of building databases. KALAKA2 and KALAKA3 [129, 130] were created for Albayzin 2010 and 2012 language recognition evaluations, respectively [123, 127]. I could describe the database building episode as some place every researcher in this field should visit once, just to realize that there is no need (nor intention) to go back again.

The NIST 2011 LRE posed new challenges, focusing on new tasks like pairwise language recognition, and promoted further research on the task, that led us to individual system development, and collaborations with other research groups [109, 113, 114, 131, 132].

The usual switching of NIST evaluations between language and speaker, kept us working on the SR field, aiming to improve our systems, adapting them to the latest techniques [46, 106–108].

Finally "The Idea" arrived: Most SLR approaches are based on either phonotactic or acoustic features. The former aims to get information from the phoneme combinations of each language, that is, tries to model the possible phone sequences allowed or present in each language and uses that information to discriminate between utterances. The latter, instead, divides the signal into short segments or frames (by using different analysis windows) and performs a frequency domain analysis, with the aim of modeling the spectral content of the signal.

SLR systems rely on combinations of these phonotactic and spectral feature based systems, as it is widely accepted that both kinds of features provide complementary information. However, it is not common practice to combine them into a single feature set, mainly because spectral features are computed on a frame-by-frame basis, whereas phonotactic features provide segmental-level information, and thus there is no clear way to mix them. Most authors build separate acoustic and phonotactic systems and fuse them at the score level to get best SLR performance [31, 133, 139, 148].

Our aim was to try to find a way to integrate phonotactic information in a frame-by-frame feature base so that the features could be modeled in state-of-the-art spectral-feature based systems.

Group meetings and brainstormings led to the Phone Log-Likelihood Ratios [50]. These features are the core of this thesis, as most of the research, experimentation and developments are performed around this set of features.

Therefore, the main objectives of the work were (or ended up being):

- Studying state-of-the-art technology for SLR and SR fields.
- Obtaining a competitive language recognition system, by development and optimization of the different modules it comprises.
- Study of the integration of spectro-acoustic and phonetic information into a new set of features.
- Introducing the new features in state-of-the-art SLR systems.
- Analyzing their performance in the most relevant benchmarks.
- Development of a competitive Speaker Recognition System.
- Integration and optimization of the features as a possible way of introducing phonotactic information in a SR system.

1.3 Structure of the Manuscript

Even though the research work with this thesis has covered both, language and speaker recognition fields, and extensive experimentation has been performed for both tasks, this thesis will focus on Spoken Language Recognition (SLR), that is, state-of-the-art technology (feature extraction techniques, modeling approaches, scoring procedures...) and all the studies will be first presented for SLR tasks. Then, the most important differences between SLR and SR tasks, and specific aspects of the latter will be outlined. The rest of this report is organized as follows:

Chapter 2 introduces state-of-the-art techniques. First, the main structure and modules of a standard recognition system are illustrated. Then, the evolution of the most relevant datasets for SLR is outlined, specifically, NIST, Albayzin and RATS benchmarks are described. Next, different kinds of features are listed, and feature extraction methodologies are analyzed. Then, the main modeling techniques and channel compensation methodologies are reported. Scoring, system backends and fusion procedures are also addressed. Finally the evaluation measures used to compare SLR system performance are defined.

The PLLR features are formally defined in Chapter 3. First, the origin of the features and the computation process is presented, as first done in [50]. An extensive study is then carried out, using the NIST 2007 LRE database, where different feature extraction parameters are optimized. Next, the performance of the proposed approach, based on PLLR features and i-vector modeling, is compared to that of several baseline systems: an acoustic approach, using the same modeling approach (i-vector) based on MFCC-SDC features, and a phone-lattice-SVM phonotactic approach, which utilizes as source of information the same phone decoders used to compute the PLLR features. This section presents results for three independent systems, based on three different phone decoders, and the analysis of the statistical significance of the results. After that, performance is studied also in other relevant datasets: NIST 2009 and 2011 LREs and Albayzin 2010 LRE, to corroborate results and explore the behavior of the features in other benchmarks and types of data [52].

The good performance attained by the system using PLLRs suggested that this set of features could be a promising characterization to use for language recognition. Still, the high feature vector dimensionality (compared to that of other frame-by-frame spectral feature vectors), could pose a problem for some modeling approaches. Chapter 4 addresses this issue by presenting a study on the reduction of dimensionality of PLLR features using supervised and unsupervised techniques [51], pursuing a lower feature dimensionality, while maintaining system performance. Then, making use of the compact representation of PLLRs, the integration of larger spectro-temporal information (also common in other spectral features) is analyzed [57].

The dimensionality reduction of the features, and the results attained with the compact representations, led to a multidimensional analysis of the feature space. Chapter 5 introduces this study, developed on NIST LRE benchmarks, revealing that PLLR features show a bounded distribution. An approach to project the features into a different space is presented, which enhances the information retrieved by the system [54, 56]. Thanks to a collaboration carried out during an internship done in the Brno University of Technology, the usage of this set of projected features is then extended to the RATS database, revealing the high performance of the features for short signals recorded in noisy environments [116].

The study is extended to speaker recognition in Chapter 6. This chapter is organized with a structure that resembles that followed in previous chapters for SLR. First, the differences between SLR and SR tasks are outlined, and state-of-the-art techniques for SR are described making emphasis on those not shared with SLR. Then, the application and usage of PLLR features on SR tasks is explained in detail. The work presented covers the optimization of PLLR features for SR and analyzes results on NIST 2010 and 2012 SRE datasets [53, 55].

Finally, conclusions and open issues to be addressed in future work are discussed in Chapter 7.

Appendix A provides details about database configuration, covering all the databases used in the experiments reported in this thesis: NIST 2007, 2009 and 2011 LRE, Albayzin 2008 and 2010 LRE, and NIST 2010 and 2012 SRE.

In Appendix B, the outcome of the participation in international challenges is outlined, including brief system descriptions and the results attained in NIST 2010 SRE [107], NIST 2011 LRE [113], NIST 2011 SRE analysis workshop [46], NIST 2012 SRE, [49], and MOBIO 2013 [84].

Chapter 2

State of the Art

This chapter first introduces the main datasets used in the SLR research area. Then the techniques employed in state-of-the-art language recognition systems are covered. The most useful and popular methods are described covering feature extraction, modeling, channel compensation, scoring, backend and fusion estimation. Finally, the main performance metrics used for system evaluation are formally defined.

2.1 Datasets

As noted before, the quantity, quality and variability of the available data for system training is of the utmost importance in the area of speech processing and language/ speaker recognition. There are various resources available for research in the spoken language recognition field. In this section, we provide details of the most relevant benchmarks for SLR.

2.1.1 NIST LRE Benchmarks

The National Institute of Standards Technology (NIST) LRE datasets [3] are, no doubt, the most relevant benchmarks for the development and evaluation of SLR technology. NIST LRE evaluations, held first on 1996 and every two years from 2003 to 2011 set not only baseline benchmarks for the research community, but have also served as reference datasets to present and evaluate emerging technologies in the SLR field. The datasets comprise low-band (8kHz) signals, mainly featuring

conversational telephone speech. In the most recent datasets (starting on 2009), data coming from Voice of America (VOA) radio broadcasts was also included (although limited to signals assessed as telephone bandwidth speech). Evaluations featured three main evaluation tracks, for different nominal speech segment durations of 3, 10 and 30 seconds.

- Starting on 1996, NIST presented a database with a set of twelve languages (where three of them included two dialects each). The dataset comprised a total amount of 20 conversations per language for training, 15 hours for development and 18 hours for evaluation (including 4560 test segments).
- The 2003 benchmark increased the amount of data with regard to that provided in 1996. The signal durations, evaluation conditions and possible evaluation tracks were the same as for the 1996 benchmark. The set of target languages was also maintained, but without dialect distinctions. The dataset comprised a total amount of 20 conversations per language for training, 30 hours for development and 3840 test segments (about 17 hours).
- The 2005 dataset [95] focused on a set of 7 languages (where two of them included two dialects each). The dataset consisted of 20 conversations for each target language. The development data consisted of the 1996 development and evaluation dataset, plus the 2003 evaluation segments. The evaluation set increased significantly, amounting to around 7550 test segments.
- In 2007, the dataset was significantly extended, increasing the number of target languages to 14, and included also several dialects, amounting to 26 different categories. Moreover, non-target language evaluation data was also provided [96] for a new open-set verification track. In this evaluation, the number of participating sites raised from the previous 12 up to 21, revealing an increasing interest on the field. All the data provided for previous evaluations was used for training or development purposes. Additionally, 20 conversations (40 conversation sides) were provided for target languages not present in previous datasets. Evaluation data amounted to 7575 test segments.
- The 2009 NIST LRE datasets increased the number of target languages up to 23. For this new benchmark, data coming from VOA radio broadcasts was also included, in addition to conversational telephone speech [93]. NIST 2009 LRE data comprised data from previous evaluations, plus VOA data, which provided non-audited training data, plus 80 30-second audited segments for each target language. Evaluation data amounted to 14059 test segments.
- The NIST 2011 LRE differed from previous evaluations in that it focused on language-pair verification conditions. That is, in previous NIST LREs, given

a speech signal, the system should give a hard decision on whether the target language was spoken in the segment considering also a set of (multiple) non-target languages. In this new task, instead, the system would solely focus on two target languages for each trial. The NIST 2011 LRE trials considered language pairs from a set of 24 target languages [69]. Training and development data for NIST 2011 LRE reused data from previous evaluations, plus a set of 100 30-second audited segments per language for languages not present in preceding datasets. The evaluation dataset amounted to 29511 test segments.

2.1.2 Albayzin LRE datasets

Following the spirit of NIST evaluations, Albayzin LREs started taking place in 2008 [125]. The Software Technologies Working Group (GTTS) [7] from the University of the Basque Country built a dataset named KALAKA [126] to support a SLR evaluation focused on the official languages in Spain. The database featured an important difference with regard to NIST 2007 LRE: speech in KALAKA was extracted from wide-band (16kHz) TV shows. The datasets of KALAKA included planned and spontaneous speech in various environmental conditions.

- KALAKA was designed to provide data to build systems for four target languages. It also provided data from other four out-of-set languages. As for NIST datasets, development and evaluation signals featured fixed durations of 30, 10 and 3 seconds. The dataset amounted to around 9 hours of training data per target language plus around 1800 speech segments for development and another 1800 for evaluation.
- KALAKA-2 [129], built for the next Albayzin 2010 LRE [128], is an extension of the previous KALAKA dataset, including two new target languages and two different training subsets of clean and noisy speech segments. The whole database amounts to 125 hours of speech, with more than 10 hours of clean speech and more than 2 hours of noisy/overlapped speech (recorded in noisy background environments) per target language. Two sets of around 1600 speech segments each were provided for development and evaluation.
- For the last Albayzin language recognition evaluation held on 2012 [123], a new application domain was selected, moving from TV broadcast speech into any kind of speech found on the Internet. In KALAKA-3 [130], data from KALAKA-2 was used for training purposes, amounting to around 108 hours of speech. Two new conditions were included, *Plenty* with six target languages for which training data was provided and *Empty* with four target languages for which no training was provided. Development and evaluation data consisted

of similar sets of between 100 and 200 audio segments (30-120 second long) per target language (plus segments for the eleven OOS languages) extracted from YouTube videos.

2.1.3 RATS

As it can be seen on the descriptions of previous datasets, while more competitive technology emerged, datasets also evolved into more challenging scenarios, involving highly confusable languages, or noisy environments in order to pose new challenges. The RATS dataset [99] was designed to perform LRE in challenging scenarios, focusing on noisy environments with evaluation tracks for speech segments of 120, 30, 10 and 3 seconds. RATS provides data for 5 target and 10 non-target languages, retransmitted through 8 different communication channels. The Linguistic Data Consortium (LDC) provided data for the RATS program, which consisted of selected signals from Callfriend and Fisher collections, previous NIST datasets and new conversational telephone speech. The data provided for training and development purposes included only signals of 120 seconds nominal duration.

2.2 Feature Extraction

In the feature extraction stage, the parameters that could better characterize the target languages are selected and extracted from the speech utterances. Feature selection and extraction is a broad and fruitful field of research in SLR. As a result, there is a wide range of features to parameterize speech.

Features can be categorized depending on several factors. A possible way of grouping them could be based on the type of information they convey, as shown in Figure 2.1 [37, 160]:

- **Phonetic, spectral and acoustic** features obtain information of the audio signals in form of speech units, that is, frame by frame based information, extracted by means of short-time (interval) windows. Languages have their own set of speech units, or phonemic inventory, which defines or summarizes the sounds covered by the language. It is, therefore, straightforward to see why the identification of the phonemes contained in an audio signal could help identifying the language spoken in an utterance. Spectral/acoustic features aim to extract information based on the principles of human sound perception.

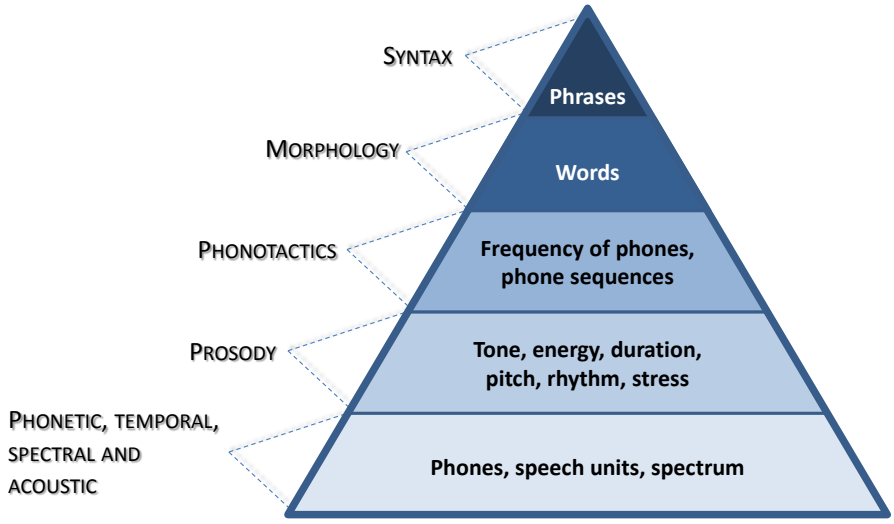


FIGURE 2.1: Classification of features for spoken language recognition.

- **Prosody** features gather information about tone, rhythm, pitch, stress, etc. These characteristics can be useful not only for affective computing (emotion recognition), but also to discriminate between languages. For instance, Isochrony can be a useful discriminative feature, as languages can divide time in equal portions at different levels, so that, some make the duration of every syllable equal, others divide time so that every mora —a unit that weights the duration of segments that conform a syllable— is equal, and others control the time between two stressed syllables. Some languages have also lexical stress rules, like the ones with *fixed stress*, which emphasize always the syllable that is placed in the same position in every word (first, penultimate...); other languages stress different syllables according to the structure of the word (*regular stress*). Tone can be not only a emotional clue, but can also be a distinctive feature to differentiate between words in tonal languages.

- **Phonotactic** features compute statistics on top of phoneme-level features, making counts of phonemes, or counts of sequences of phones (n-grams), building this way characteristics covering a larger temporal context. The usefulness of these features can be easily guessed, given that even languages which could share a phonetic inventory would have different allowed phoneme combinations to build words.

- **Morphology** features refer to the analysis of morphemes and lexemes and the way these are used to build words in languages. Building a system which is able to identify the words contained in an audio signal would make easy

the language recognition task, but at a high cost, given that large language dependent dictionaries would be needed for these approaches to be applied.

- **Syntax** features study the way phrases are built out of words. Some tasks require the use of syntax-based features, such as large vocabulary continuous speech recognition.

All these features provide benefits for language recognition, yet, the most popular approaches combine spectral and phonotactic features, given that these features are not only easy to extract compared to prosodic, morphological or syntax-based features, but also attain good performance. Furthermore, commonly referred to as *low-level* and *high-level* features, they have shown to carry complementary information [22] [133].

Several works have also explored the use of prosody and other features [76] [103] [97]. These systems, though not being that competitive by themselves, provide complementary information to the systems based on acoustic or phonotactic features.

In this section, we will give an overview of the feature extraction stage, starting with signal processing basics, giving some details about the voice activity detection stage and providing details about the extraction processes of the most common low and high level features for spoken language recognition.

2.2.1 Signal processing

Signal processing is a field of research that studies the techniques used to process, transform and analyze all kind of physical signals (acoustic, electromagnetic, etc.). The first stage of any language, speaker or speech recognition system is the conversion of the input utterance into a sequence of parameters with relevant information for the recognition task, which is based on different speech audio signal processing techniques.

Speech processing states that some speech properties are short-time varying, or quasi-stationary in short time intervals (of about tens of miliseconds) [13]. Besides, it is also known that the human auditory system analyses signals in the frequency domain. Many feature extraction methods, therefore, rely on *short time analysis* to study signals, making use of frames or windows through the signal to extract information. This process is known as signal sampling or windowing. After that, the signal is transferred into the frequency domain by means of the Discrete Fourier Transform (DFT).

The choice of the window length (typically from 20 to 30 ms. for speech applications) is a compromise between the stationarity assumption and the frequency resolution attained afterwards. The window shift is usually chosen to obtain overlapped windows (with frame rate of around 10 ms.) to avoid the information loss between frames. Some other issues arise as a consequence of using short analysis windows: The DFT assumes that the analyzed signal is periodic in the considered frame. Computing the DFT of signals that are non-periodic in the analysis window causes an effect known as leakage, which distorts the frequency components of the signal. Different types of windows have been studied in the literature with the aim of optimizing the spectral analysis of signals, reducing leakage while maintaining a low variance. Hamming is the window most commonly applied, though many others have been explored: Dirichlet, triangle, Hanning, Blackman, etc. [71, 115].

Another important concept in speech processing is the so called *Cepstrum*, which is the spectrum of the logarithm of the power spectrum of a signal [13]. The application of the logarithm in the frequency domain provides a smoothed representation of the signal spectrum, making it possible to characterize the articulation (vocal tract configuration) of the produced sound. DFT components are typically averaged in a bank of filters scaled according to a perceptual scale, so that a set of perceptual-frequency energies are obtained. Finally, a Discrete Cosine Transform (DCT) is applied in order to suppress correlations among filter energies. Cepstrum is broadly used to characterize speech, as it provides a way to separate in the frequency domain the components of the speech signal corresponding to (voiced or unvoiced) excitation and vocal tract filter response (which is the relevant information).

2.2.2 Voice Activity Detection

Voice Activity Detection (VAD) is an important and challenging process at the feature extraction stage, that deals with the problem of discriminating between speech and non-speech (silent or noisy) regions of the audio signals. An ideal VAD should be robust for a wide range of Signal to Noise Ratio (SNR) conditions and for different types of noises: white, stationary, impulses, etc.

VAD can be performed in different ways. The simplest techniques are based on the energy content (at the frame level) of speech signals. More sophisticated VAD techniques use other frame-level characteristics, like zero crossing rate, cepstral features, formant shape, or even phonetic features as in [18, 77], where a phonetic recognizer was used to identify silence or noisy frames in order to discard them. Speech and noise statistical models [35, 140], or long-term signal analysis methods [67] have also been widely explored.

2.2.3 Spectral and Acoustic *low-level* Features

Low-level features model languages with information taken from quasi-stationary spectral characteristics of the audio signal, and their evolution over time. These features are easy to extract and provide a good characterization of the language, making them the most used features in the literature and in applications.

There is a considerable amount of spectral and acoustic features that have been used for speech and/or speaker recognition, but that have not been widely explored for language recognition tasks: gammatone frequency cepstral coefficients, linear frequency cepstral coefficients, frequency domain linear prediction, etc. In [158] some of these features are collected and tested in a SLR system, and some further innovative features or variations of the previous are also presented. In this Section, we will describe the most used features in SLR literature.

Mel Frequency Cepstral Coefficients

Among acoustic features, the Mel-Frequency Cepstral Coefficients (MFCC) are one of the most common representations for language, speaker and speech recognition [18, 139]. The MFCC feature computation was first proposed in [38]. This feature extraction procedure averages the spectral energies in frequency bands defined according to the human perception of differences in frequency, the so called Mel scale. The Mel scale defines an approximately linear scale for frequencies below 1000 Hz, and a logarithmic scale for frequencies above it. This way, a higher resolution is provided for low frequencies.

The full extraction process can be described as follows: First, audio signals are sampled using typically 20-30 ms Hamming windows at a 10 ms rate. Next, a DFT (in fact, a Fast Fourier Transform, FFT) is applied to get the representation in the frequency domain. Right after, Mel filtering [78] is applied to get the spectral energies around 20 Mel-scaled frequency bands and logarithms of the output of the filters are computed. Finally a DCT is applied, which provides the representation known as MFCC.

Dynamic coefficients

Several works have shown empirically that the use of information relative to the evolution of acoustic features is effective in language recognition. Therefore, dynamic coefficients are usually computed on top of MFCC features.

First-order dynamic coefficients are computed as defined in [156]:

$$\Delta f(t) = \frac{\sum_{d=1}^{D} d\,[f(t+d) - f(t-d)]}{2\sum_{d=1}^{D} d^2} \tag{2.1}$$

where $f(t)$ is a feature at time t, and $2D+1$ is the size of the regression window. Second-order dynamic coefficients ($\Delta\Delta$) are computed using Eq. 2.1 on first-order dynamic coefficients.

Shifted Delta Cepstrum

SDCs provide another way of making use of larger temporal context information in the features. Unlike MFCC+Δ+$\Delta\Delta$ feature vectors, which carry information of static, first and second order dynamic characteristics, SDCs carry only static and first order dynamic information, computed over a certain number of surrounding windows, centered on the analysis window. That is, SDCs characterize the language by the evolution of local variations of the spectrum around the analysis window. Shifted Delta Cepstrum (SDC) computed on top of MFCCs, is also state-of-the-art in language recognition [18, 139, 149]. SDCs are specified by four parameters N-d-P-k [149]: N is the number of coefficients from which derivatives are computed at each frame, d determines the size of analysis windows (consisting of $2 \cdot d + 1$ frames) to compute the derivatives, P is the shift (number of frames) between two consecutive analysis windows and k is the number of analysis windows whose delta coefficients are concatenated to form the final feature vector. Figure 2.2 shows a diagram of the parameters for SDC feature computation.

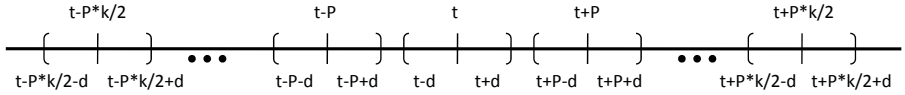


FIGURE 2.2: Window parameter definitions for the estimation of the first order deltas that conform SDC features.

After the computation of dynamic coefficients, frames marked by the VAD module as silence or non-speech are removed.

Perceptual Linear Prediction Features

Another common characterization (specially in speech recognition tasks) is the one consisting of Perceptual Linear Prediction features (PLP) [73], which are based

on several concepts related to the critical band masking property of the human auditory model in order to estimate the signal spectrum. Given an audio signal, the PLP estimation algorithm first computes the power spectrum, then warps the frequency axis using the Bark scale. Next, it performs a convolution with a critical-band masking curve and down-samples the signal, preemphasizes the result and applies an intensity-loudness warping. Finally, auto-correlation method of all-pole spectral modeling is applied, which gives autoregressive coefficients, that can be further transformed to obtain parameters like cepstral coefficients.

2.2.4 Phonetic and Phonotactic *high-level* Features

Unlike spectral features, high level features are more robust against channel and noise variabilities. However, they are not easy to compute, and large amounts of data are necessary to estimate them. Among high level features, the most common representations are based on the information provided by phone decoders [66] [26] [146] [112].

Phone decoders

Phone decoders need a large amount of labeled data (determining the phonemes that are present in the speech) to train the models for each of the phones of their phonetic inventory [26]. Once they are trained, phone decoders are able to take an input sequence of acoustic observations X, and provide an acoustic posterior probability of each state s ($1 \leq s \leq S$) of each phone model i ($1 \leq i \leq N$) at each frame t, $p(i|s,t)$. This phone-state posterior probabilities are later used to obtain 1-best phone decodings, or lattices (see below).

Phone decoders can rely on different features, like the MFCCs or PLPs introduced above, usually augmented with Dynamic coefficients or other techniques to obtain information for larger temporal contexts [66, 137]. Once the features are obtained, the input frames are scored with regard to the phonetic units of the phonetic inventory. This can be done using different modelings: GMMs, HMMs, Neural Networks or hybrid models are commonly used in this step.

Phonemes or speech units, usually span several frames. The scoring at the beginning and end of each speech unit is usually worse than the one attained in the middle frames. With the aim of improving the acoustic modeling of the phonemes, phone decoders are usually trained to provide phone-state posteriors, which represent shorter units and allow a better acoustic modeling of the speech.

BUT TRAPs/NN phone decoders

Some of the phone decoders more used in the literature are the open software Temporal Patterns Neural Network (TRAPs/NN) phone decoders, developed by the Brno University of Technology (BUT) [137]. These phone decoders rely on multilayer perceptron NN, and have the following structure:

First audio signals are sampled using 25ms Hamming windows at a 10 ms rate. Mel filter bank energies are then estimated, and TRAP feature vectors are computed for each critical band, which module the evolution of the energy values. Mean and variance normalization is applied to the vector, which is then Hamming-windowed, and normalized again with regard to the training set. Each of these vectors is the input to a NN classifier, which estimates phone posterior probabilities for each phone-state model and critical band. The outputs of the classifiers feed another NN, known as *merger*, which combines the outputs into a single posterior probability vector per frame.

The BUT TRaPs/NN phone decoders were developed for Czech (CZ), Hungarian (HU) and Russian (RU), feature 45, 61 and 52 phonetic units respectively, and provide three posterior probabilities per phone unit and frame, that is, they are implicitly using three-state phonetic models.

The main training features of these decoders are:

- Czech Decoder (CZ) - 8 kHz, trained on the Czech SpeechDat(E) Database, containing 12 hours of speech from 1052 Czech speakers (526 males, 526 females), recorded over the Czech fixed telephone network.
- Hungarian Decoder (HU) - 8 kHz, trained on the Hungarian SpeechDat(E) Database, containing 10 hours of speech from 1000 Hungarian speakers (511 males, 489 females), recorded over the Hungarian fixed telephone network.
- Russian Decoder (RU) - 8 kHz, trained on the Russian SpeechDat(E) Database, containing 18 hours of speech from 2500 Russian speakers (1242 males, 1258 females), recorded over the Russian fixed telephone network.

1-best Decoding

In the early times of phone decoder based approaches, given an utterance, a phone decoder (trained on one or more languages) would provide the most likely phone sequence according to its phonetic inventory, that is, the most likely sequence of phones.

These sequences are then normally used to build n-grams, that is, sequences of N phones and counts of n-grams are used as features:

$$count(\hat{w}_i, w_i|\Theta^*) \tag{2.2}$$

where $\hat{w}_i = w_{i-(n-1)}, ..., w_{i-1}$, and w_i is the phone at position i of the optimal sequence of phones Θ^*.

Phone Lattices

In more recent phone decoder based approaches, phone lattices were introduced. Lattices model speech utterances as graphs, where nodes represent points in time and each arc corresponds the a phone hypothesis and its corresponding acoustic score. With lattices, instead of just taking the most likely phone sequence, a summation is performed over the different phone sequences [66]. Figure 2.3 shows the representation of a phone lattice and the 1-best decoding output for a single utterance.

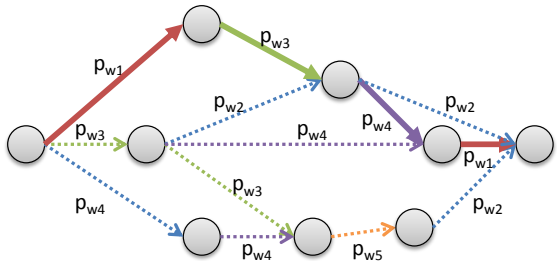


FIGURE 2.3: Representation of a phone lattice of a utterance using a phone decoder of 5 phone units. The 1-best decoding output is marked as the optimal path in the lattice.

It has been experimentally shown that lattices provide a better way of estimating the n-gram probabilities than the 1-best decoding, that is, the information retrieved by summing the likelihoods over all the paths in the phone lattice is more robust than that obtained from the best path.

Aiming to maximize the information carried out by the features, works in the literature have widely explored different n-gram orders. The increment in the n-gram order enhances system performance, as the extracted higher order n-grams carry more language-specific information, but entails a higher computational cost (as the number of features grows exponentially with n) which may make the modeling of the

features intractable. Techniques dealing with this fact normally use n-gram selection methods, discarding low-frequency n-grams (based on counts made on training sets) [121].

2.3 Modeling

Modeling techniques can be classified with regard to different criteria. Based on the training methodology, models can be classified as either generative or discriminative. The former estimates intra-class variability whereas the latter searches for the boundaries between classes. Besides, models can be also seen as parametric or non-parametric. In parametric models, each class is assumed to be modeled by a specific probability density function and model parameters are estimated so as to best fit the probability distribution. Non-parametric models, instead, compare feature vectors directly and the degree of similarity between them is used as a metric to decide whether they belong to the same class or not.

In this section, we will present the modeling techniques for language recognition that stand out as more popular in the literature.

GMM-UBM

In low-level acoustic systems, the target language is modeled with information taken from the spectral characteristics of the audio signal. In the early times of acoustic-based approaches for language recognition, though these systems provided advantages in terms of computational complexity, performance was not as good as that attained by phonotactic systems [159]. The approach known as Gaussian Mixture Model / Universal Background Model (GMM-UBM) previously used in speaker recognition systems [120], was first introduced for language recognition in [155] using Mel-Frequency Cepstral Coefficients as features.

GMM are widely used in speech, speaker and language verification systems. A GMM is a generative model that represents the acoustic characteristics of the training set by means of a probability density function defined as a weighted sum of Gaussians, that are defined in the vector space of the acoustic parameters. A GMM consists of a mixture of K D-dimensional Gaussians. A vector $\mathbf{w}$ defines the weight of each Gaussian component in the mixture, and each Gaussian is modeled by a vector of means $\boldsymbol{\mu}$ and a $D \times D$ covariance matrix $\boldsymbol{\Sigma}$. The probability of the input feature vector $\mathbf{f}$, for a GMM model $\mathcal{M}$, is given by:

$$p\left(\mathbf{f}|\mathcal{M}\right) = \sum_{k=1}^{K} w_k \frac{1}{\left(2\pi\right)^{D/2}\left|\mathbf{\Sigma}_k\right|} \exp\left\{-\frac{1}{2}\left(\mathbf{f}-\boldsymbol{\mu}_k\right)^T \mathbf{\Sigma}_k^{-1}\left(\mathbf{f}-\boldsymbol{\mu}_k\right)\right\} \tag{2.3}$$

There are different approaches and algorithms to estimate the parameters of these statistical models. Among them, maximum likelihood based Expectation Maximization (EM) [42] has been widely used in the literature.

The EM algorithm aims to maximize the likelihood of the observed data set with regard to the different parameters (means, covariances and weights of the components) of the model $\mathcal{M}$. For that purpose, the parameters are first initialized (either randomly, or using the K-means algorithm or another reasonable approach). In the E step, the algorithm estimates the posterior probabilities of the responsibilities given the parameters, that is, it estimates the responsibility that each component k takes for generating the data. Then, in the maximization step, the model parameters are reestimated with regard to those responsibilities. Finally the log-likelihood is evaluated to check if it has reached the convergence criterion; if not, the E-M steps are repeated until convergence or a certain number of iterations are reached.

In the GMM-UBM approach, a universal GMM is trained with data involving a relatively high number of spoken languages (which may or may not include target languages), to get a model that covers all the variability that we may expect in the input utterances of the SLR application. Models of target languages are then trained either by EM, or by Bayesian adaptation, Maximum A Posteriori (MAP) [65, 120, 155], which presents the advantage of requiring less training data for each target model.

MAP consists of adapting an universal model to the selected data —a specific target language in SLR—. The algorithm combines the parameters of the UBM with the ones estimated from the new data by means of a relevance factor. The factor determines the degree in which the parameters will be transformed into the new ones: a big learning factor would require a higher exposure to parameters in the new data, for the component of the UBM to be adapted. MAP adaptation can retrain means, weights and covariances [120]. In most approaches only the UBM mean vectors are adapted for each target model. First, training samples are used to compute mean and weight expectations:

$$n_k = \sum_{t=1}^{T} P(k|\mathbf{f}_t, \lambda) \tag{2.4}$$

$$E_k(\mathbf{f}) = \frac{1}{n_k} \sum_{t=1}^{T} P(k|\mathbf{f}_t, \lambda)\mathbf{f}_t \tag{2.5}$$

where $\mathbf{f}_t$ is the input feature vector, λ are the parameters of the GMM, and P denotes the posterior probability (this estimation is the same in the E step on the EM algorithm). Mean adaptation is performed following:

$$\hat{\boldsymbol{\mu}}_k = \alpha_k E_k(\mathbf{f}) + (1 - \alpha_k)\boldsymbol{\mu}_k \tag{2.6}$$

where α_k is the adaptation coefficient, defined as:

$$\alpha_k = \frac{n_k}{n_k + r} \tag{2.7}$$

where r is the relevance factor.

The concatenation of MFCC-SDC features in [149] under the GMM-UBM modeling approach proved to be a successful way of making use of acoustic information for SLR.

Support Vector Machines

In the mid 2000 years, Support Vector Machine (SVM) modeling was introduced for acoustic SLR [30] using MFCC-SDC as features.

Support Vector Machines, unlike GMMs, are discriminative models that try to find the boundaries between two classes. SVMs search for the separating hyperplane $h(x)$ defined as:

$$h(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i c_i K(\mathbf{x}, \mathbf{x}_i) + d \tag{2.8}$$

where $K(\cdot, \cdot)$ is a kernel function; $c_i = \pm 1$ corresponds to the class label; α_i are weights so that $\sum_{i=1}^{N} \alpha_i c_i = 0$ and $\alpha_i > 0$; $\mathbf{x}_i$ are the support vectors obtained from

the training data and d is a learning factor. SVMs must consider that classes might not be linearly separable. For that reason, the input data is mapped into a high dimensional space by $\mathbf{b}(\mathbf{x})$. The Kernel must satisfy Mercer's condition, which can be expressed as $K(\mathbf{x}, \mathbf{y}) = \mathbf{b}(\mathbf{x})^t \mathbf{b}(\mathbf{y})$. The value of $h(\mathbf{x})$ above or below a given threshold will determine the class that corresponds to the input data.

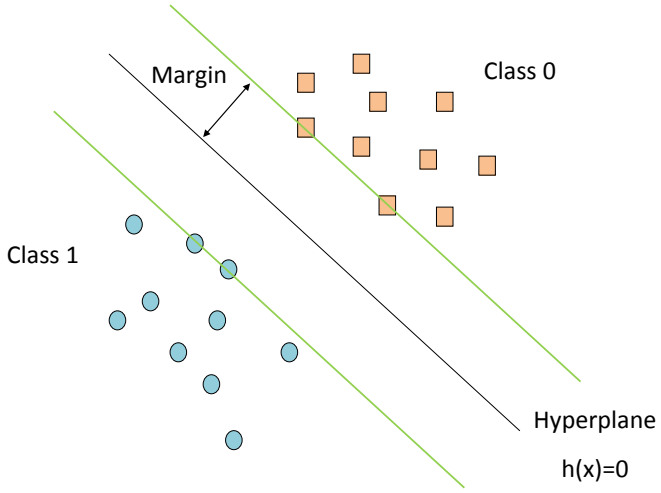


FIGURE 2.4: Two dimensional representation of a SVM

The SVM searches for the hyperplane that maximizes the margin between classes. Geometrically, the margin is the smallest distance between the training data of each class and the hyperplane (see Figure 2.4). Therefore, the metric used in each case will condition the solution found for the hyperplane.

GMM-SVM

In [27], GMM supervectors were introduced for a SVM-based speaker recognition system, and [33] extended this work for spoken language recognition. GMM supervectors are built by concatenating Baum-Welch statistics which map the utterances into a high dimensional space.

Given a GMM $\mathcal{G} \equiv \{\omega_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k | k = 1..K\}$ consisting of K Gaussians in a D-dimensional space, with diagonal covariance matrices $\boldsymbol{\Sigma}_k$, the zero order statistics are computed as:

$$n_k = \sum_k \gamma_k(t) \tag{2.9}$$

where $\gamma_k(t) = P(k|\mathbf{f}_t, \mathcal{G})$ is the posterior probability of the component k of the GMM, given the parameters and the feature vector $\mathbf{f}_t$ at time t.

The first order statistics are defined as follows:

$$\mathbf{x}_k = \sum_k \gamma_k(t) \boldsymbol{\Sigma}_k^{-\frac{1}{2}} (\mathbf{f}_t - \boldsymbol{\mu}_k) \tag{2.10}$$

the parameter vectors $\mathbf{n} = [\overbrace{n_1, \ldots, n_1}^{D}, ..., \overbrace{n_K, \ldots, n_K}^{D}]'$ and $\mathbf{x} = [\mathbf{x}_1, \ldots, \mathbf{x}_K]'$ are known as the zero and first order sufficient statistic supervectors, respectively. In the GMM-SVM approach, these MAP-adapted GMM supervectors are used to train the SVM classifiers.

Phonotactic approaches

Assuming 1-best decoding and given L models for L target languages and the most-likely sequence of phones Θ^* provided by a phone decoder, this approach aims to find the target language which maximizes [29]:

$$L^* = \underset{L}{\text{argmax}} \log p(\Theta^*|L) \tag{2.11}$$

$$= \underset{L}{\text{argmax}} \frac{1}{N} \sum_{i=1}^{N} \log p(w_i|w_{i-(n-1)}, ..., w_{i-1}, L) \tag{2.12}$$

where $w_1, ..., w_N$ are the N phones contained in Θ^* and n is the n-gram order. The joint probability of the sequence of phones can be also expressed in terms of n-gram counts as follows:

$$p(\hat{w}_i, w_i|\Theta^*) = \frac{count(\hat{w}_i, w_i|\Theta^*)}{\sum_{j=1}^{J} count(\hat{w}_j, w_j|\Theta^*)} \tag{2.13}$$

where J are all the unique n-grams in the utterance. Therefore, using this notation, under the 1-best decoding, this approach aims to find the language that maximizes:

$$L^* = \underset{L}{\text{argmax}}\ s_L(\Theta^*) \tag{2.14}$$

$$s_L(\Theta^*) = \sum_i p(\hat{w}_i, w_i|\Theta^*)\ \log\ p(w_i|\hat{w}_i, L) \tag{2.15}$$

In the case of using phone lattices, the summation involves many different unique $\hat{w}w$ n-grams for which the expected likelihood is computed as an average over all possible paths in the lattice [29]:

$$E_{\mathcal{W}}[s_L(\Theta)] = \sum_{\hat{w}w} E_{\mathcal{W}}[p(\hat{w}, w|\Theta)]\ \log\ p(w|\hat{w}, L) \tag{2.16}$$

Systems using phone decoders as feature extractors have relied on different approaches, either using a single Phone Recognizer followed by Language Modeling (PRLM) (see Figure 2.5) or using multiple Parallel Phone Recognizers followed by Language Modeling (PPRLM), in which several language dependent phone recognizer based systems are fused, as shown in Figure 2.6.

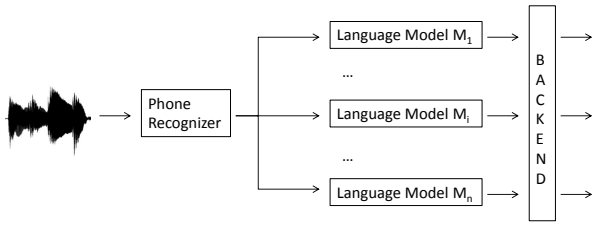


FIGURE 2.5: Diagram of a PRLM system

Phone-lattice-SVM

SVMs have also been used under the approach known as *Phone-lattice-SVM*, combining phonotactic features with SVM classifiers [26].

The phone lattice produced by a decoder i is stored for each target language l_j, then feature vectors are built from expected counts of phone n-grams. A SVM model $\psi(i, l_j)$ is estimated on the outputs of the phone decoder i for the training dataset, taking j as the target language. Variations of this approach have been recently proposed leading to improved performance [112, 146].

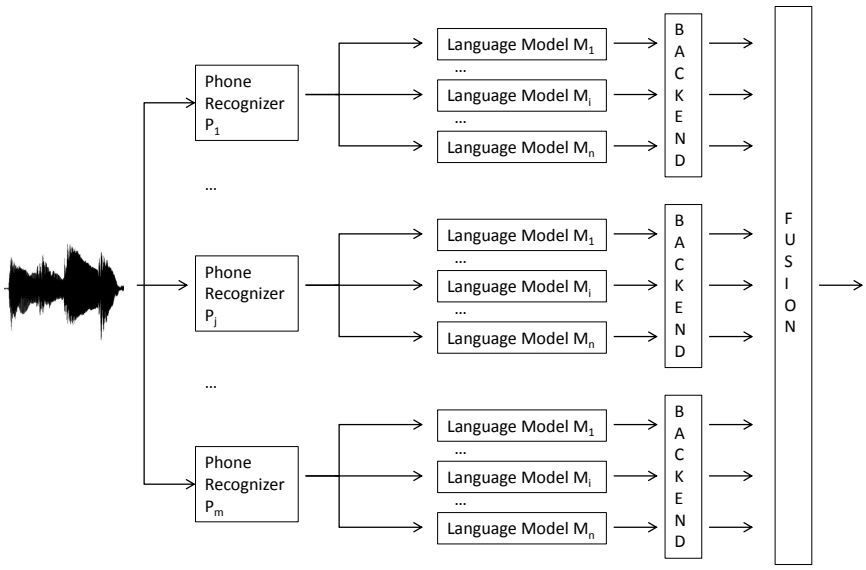


FIGURE 2.6: Diagram of a PPRLM system

At the feature level, several methods have dealt with dimensionality reduction issues to try to decrease the feature n-gram order using different criteria depending on SVM-modeling parameters, feature discrimination merits, projections, or dynamic selections [110, 121, 147].

2.4 Channel Compensation

The term *channel compensation* comprises all the techniques that try to suppress or minimize the undesired variabilities contained in speech signals. In the case of spoken language recognition tasks, these variabilities are caused (among others) by environmental noises, variabilities introduced by different recording devices, transmission channels or speakers, and are usually summarized as channel variabilities.

Channel variabilities pose nowadays one of the main difficulties for system performance. Eliminating the undesired variabilities while keeping the desired features for language modeling is a complicated task, which is addressed at different stages of the recognition system. This section describes the main state-of-the-art techniques applied to solve channel variability issues.

Feature-level channel compensation

The main disadvantage of spectral features is their sensitivity to noise and channel variabilities. Several compensation techniques have been applied, to minimize these undesired effects.

Cepstral Mean Subtraction (CMS) deals with slow varying distortions, like stationary noises and the effects caused by the use of stationary filters, like the type of microphone, the distance to the microphone and the acoustics of the area where the recording is made. The technique consists on computing the mean of each cepstral coefficient on the signal file and subtracting it from the corresponding coefficient [89, 136]. Another variability compensation technique applied on the feature extraction stage (more common in speaker recognition) is RASTA filtering [74]. The technique assumes that channel characteristics have little frequency variations over time, so their spectral elements are in the low-frequency area. RASTA applies a bandpass filtering to each frequency channel to get rid of those variabilities. Another commonly applied technique, called Feature Warping [104], maps the parameter probability density function to a normal distribution, which makes the features more robust to linear channel effects.

Nuisance Attribute Projection

A family of compensation techniques is based on the idea that channel mismatch, environmental noise and other undesired variabilities are contained in a lower dimensional subspace. Methods based on this hypothesis use labeled data (with either language or condition dependent labels) to estimate the different distortions or channel variabilities, and then project features or compensate the models in order to remove those undesired effects.

Nuisance Attribute Projection (NAP) [141] was one of the first methods proposed based on this idea. NAP estimates the channel factors on the SVM or supervector spaces, and projects them out leading to features that are more resistant to channel effects [122].

NAP requires data from several sessions for each language. Given the language l and a data matrix with feature vectors $\mathbf{f}$ recorded over different conditions, NAP tries to minimize the following function:

$$F(\mathbf{P}) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} W_{ij} ||\mathbf{P}(\mathbf{b}(\mathbf{f}_i) - \mathbf{b}(\mathbf{f}_j))||^2 \tag{2.17}$$

where N is the number of input signals, $\mathbf{W}$ is a matrix where $W_{ij} = 1$ if $\mathbf{f}_i$ and $\mathbf{f}_j$ belong to the same language, and 0 when feature vectors belong to different languages, and $\mathbf{P}$ is the projection matrix that is assumed to follow the form:

$$\mathbf{P} = \mathbf{I} - \mathbf{X}\mathbf{X}^t \tag{2.18}$$

Eigenchannel Compensation

After NAP, channel compensation techniques at the modeling stage started to gather strength in SLR. Methods previously applied to speaker recognition like eigenchannel adaptation [82] were afterwards successfully applied to spoken language recognition [32] [75] [34] [21].

Under this approach, eigenchannels are estimated and used to move the supervectors in the maximum variability direction, with the aim of adapting the models to the test data.

In order to perform eigenchannel compensation, first, an uncompensated supervector is computed for each utterance:

$$\mathbf{m} = (r\mathbf{I} + diag(\mathbf{n}))^{-1}\,\mathbf{x} \tag{2.19}$$

where $\mathbf{I}$ is the identity matrix, $diag(\mathbf{n})$ is a matrix of dimension $D \times K$ with the supervector $\mathbf{n}$ in the diagonal and r is an heuristic relevance factor.

Language variabilities (not willing to be modeled) are removed from the supervectors by computing the mean of the training signals belonging to the same language. Then the covariance matrix of the training samples is used to compute the most relevant Q eigenvectors by means of Principal Component Analysis (PCA). Then the eigenchannel matrix $\mathbf{W}$ is built by concatenating each eigenvector $\mathbf{v}_i$ multiplied by its corresponding eigenvalue λ_i.

$$\mathbf{W} = [\mathbf{v}_1 \cdot \sqrt{\lambda_1}\,\mathbf{v}_2 \cdot \sqrt{\lambda_2} \ldots \mathbf{v}_Q \cdot \sqrt{\lambda_Q}] \tag{2.20}$$

When data is labeled with regard to the type of recording or transmission channel, different eigenchannels can be estimated depending on the channel variability to be modeled. In that case, the final $\mathbf{W}$ matrix is formed by the concatenation of the different sub-matrices.

Once the eigenchannels are computed, these can be used to perform the compensation in the sufficient statistics space, as follows:

$$\hat{\mathbf{x}} = \mathbf{x} - diag(\mathbf{n}) \cdot \mathbf{W}\mathbf{L}^{-1}\mathbf{W}^t\mathbf{x} \tag{2.21}$$

where $\mathbf{L}$ is defined as:

$$\mathbf{L} = \mathbf{I} + \mathbf{W}^t diag(\mathbf{n})\mathbf{W} = \mathbf{I} + \sum_{k=1}^{K} n_k \cdot \mathbf{W}_k^t \cdot \mathbf{W}_k \tag{2.22}$$

Finally, the compensated supervectors are computed by:

$$\hat{\mathbf{m}} = (r\mathbf{I} + diag(\mathbf{n}))^{-1}\,\hat{\mathbf{x}} \tag{2.23}$$

These compensated supervectors can be then either used to feed a SVM, or directly applied for linear scoring or used to feed other classifiers.

Joint Factor Analysis

The channel compensation approaches presented so far focused on modeling the channel-related part of the speech signal/utterance and *removing* it from the utterance.

The Joint Factor Analysis (JFA) approach [80], originally introduced for speaker recognition technology, was successfully applied to spoken language recognition too.

Joint Factor Analysis took a step forward compared with previous approaches, introducing, along with the supervector models, two channel and language dependent factors as follows:

$$\mathbf{M} = \mathbf{m}_{ubm} + \mathbf{U}\mathbf{x} + \mathbf{V}\mathbf{y} + \mathbf{D}\mathbf{z} \tag{2.24}$$

where $\mathbf{U}$ is the channel subspace, $\mathbf{V}$ is the language subspace and $\mathbf{D}$ is a diagonal matrix covering the residual variability. The vectors $\mathbf{x}$ and $\mathbf{y}$ are the channel dependent and language dependent factors, respectively, and $\mathbf{z}$ is the residual factor; all of them are assumed to be normally distributed.

Data from various channel conditions and languages is needed to estimate the model parameters. The language factors are forced to be the same for each language, whereas different channel factors are estimated on each utterance. Channel factors are also estimated on the test step, and used to adapt the language model to the test channel condition.

Total Variability Factor Analysis

Finally, JFA gave rise to *Total Variability Factor Analysis*. This approach arose as a result of a set of experiments which proved that, based just on the channel components of the JFA approach, it was possible to perform SR with recognition rates much better than random. The discovery suggested that the channel components were therefore carrying speaker-related information, that was lost in the speaker subspace. Under the Total Variability Factor Analysis approach, all the information is stored in a single feature vector, by projecting data into a low-dimensional subspace. In this way, all the relevant information is conveyed by low-dimensional feature vectors known as *i-vectors*, which can be then processed using further modeling approaches [40] [41] [98].

Under the total variability modeling approach [40], an utterance dependent GMM supervector $\mathbf{M}$ (stacking GMM mean vectors) is decomposed as follows:

$$\mathbf{M} = \mathbf{m} + \mathbf{T}\mathbf{w} \tag{2.25}$$

where $\mathbf{m}$ is the utterance independent mean supervector, $\mathbf{T}$ is the total variability matrix (a low-rank rectangular matrix) and $\mathbf{w}$ is the so called *i-vector* (a normally distributed low-dimensional latent vector). That is, $\mathbf{M}$ is assumed to be normally distributed with mean $\mathbf{m}$ and covariance $\mathbf{T}\mathbf{T}'$. The latent vector $\mathbf{w}$ can be estimated from its posterior distribution conditioned to the Baum-Welch statistics extracted from the utterance and using a UBM. The i-vector approach maps high-dimensional input data (a GMM supervector) to a low-dimensional feature vector (an i-vector), hypothetically maintaining most of the relevant information.

Then, i-vectors can be used to feed different classifiers from simple generative modeling approaches (a single Gaussian to model each target language) or logistic regression to more complex approaches, like feeding neural networks, or training a Probabilistic Linear Discriminant Analysis model.

2.5 Scoring

In SLR, given a trial, consisting of a test segment S and a target language l_T, the system must decide whether or not the target language is the language spoken in the test segment. The decision is taken according to the value of the score attained by the signal against the model of the language $s(S, l_T)$ with regard to a threshold θ. The trial will be accepted only if $s(S, l_T) > \theta$.

The scores can be computed in different ways. The outputs of some of the presented modeling approaches can be directly treated as scores (NN, PLDA, PPRLM, etc.) Some other modelings need further processing to transform the outputs into reliable scores. For example, supervectors or i-vectors can be tested against each other using cosine distance, or they can feed other classifiers such as the mentioned SVMs, logistic regression or NNs to get scores.

In most cases, scores still need further processing, either a fusion step to combine the outputs of several systems or a calibration step also known as backend.

2.6 Calibration and Fusion

Calibration of the scores is required to set a single (general) threshold θ on which the accept/reject decisions rely, that is, scaling the scores for the different language models equalizing them for the whole set. Calibration is also applied in order to produce meaningful scores (e.g. posteriors) [22]. The calibration stage can involve different techniques, like normalization, backend models or the fusion of several systems.

2.6.1 Score Normalization

Score normalization techniques such as Z-norm and T-norm [12] can help removing the environmental effects on the score space. Nevertheless, they are rarely applied alone in SLR systems. Instead, they are usually applied before some other backend.

Z-Norm

The Z-norm aims to compensate for deviations related to the target language. Given two language models: an odd model m_1 (for language l_1) given the characteristics of its training signals, and a "standard" model m_2 (for language l_2), if the system was

to evaluate a clean signal S_1, containing speech in language l_1 against both models, the system could assign a low score to S_1 on its evaluation against l_1, compared to the score attained against l_2. To minimize this effect, scores for each language are normalized with regard to a set of development signals containing non-target languages, as follows:

$$Zscore = \frac{s_l - \mu_z(m)}{\sigma_z(m)} \tag{2.26}$$

where $\mu_z(m)$ and $\sigma_z(m)$ are the mean and standard deviation of the scores of model m against the Z-norm set.

T-Norm

The T-norm aims to compensate for deviations related to the test signal. Suppose that S_1 contains speech in language l_1 and S_2 speech from some other language. If S_1 contained singularities (noises, laugh, screams, etc.) and S_2 contained normal speech, the score of S_1 against l_1 would probably be low with regard to the one attained by S_2 against l_1. To get rid of this effect, the T-norm normalizes test-signal scores using a set of non-target models following:

$$Tscore = \frac{s_l - \mu_t(m)}{\sigma_t(m)} \tag{2.27}$$

where $\mu_t(m)$ and $\sigma_t(m)$ are the mean and standard deviation of the scores of model ml against the T-norm set.

ZT-Norm

The score combining both Z-norm and T-norm would be computed as:

$$ZTscore = \frac{\frac{s_l - \mu_z(m)}{\sigma_z(m)} - \mu_t(y)}{\sigma_t(y)} \tag{2.28}$$

where $\mu_z(m)$ and $\sigma_z(m)$ are the mean and standard deviation of the scores of model ml against the Z-norm set and $\mu_t(y)$ and $\sigma_t(y)$ are the mean and standard deviation of the Z-normalized score yl against the T-norm set.

2.6.2 Backend models

The backend serves as a precalibration stage that transforms the space of scores to get reliable estimates of the class probabilities. Besides, when the set of languages for which models have been trained does not match the set of target languages, the backend maps the available scores to the space of target languages.

In the case of NIST LRE datasets, separate models can be trained for different dialects of a target language or for different data sources (telephone conversational speech, radio broadcast speech, etc.), and non-target languages can be modeled as well.

Several kinds of backends can be applied [109]. In this section we will just outline the ones mostly applied in state-of-the-art systems.

Generative Gaussian Backend

In a generative Gaussian backend, the distribution of language scores is modeled by a multivariate normal distribution $\mathcal{N}(\mu_{l_T}, \mathbf{\Sigma})$ for each target language l_T, where the full covariance matrix $\mathbf{\Sigma}$ is shared across all target languages. Maximum Likelihood (ML) estimates of the means and the covariance matrix are computed.

Given a score vector $\mathbf{s}$ of size K, the output (calibrated) log-likelihood vector $\hat{\mathbf{s}}$ is obtained by:

$$\hat{\mathbf{s}} = \mathbf{A}\mathbf{s} + \mathbf{b} + \mathbf{c} \tag{2.29}$$

where the rows of $\mathbf{A}$ are:

$$\mathbf{a}_{l_T} = \mu'_{l_T}\mathbf{\Sigma}^{-1} \tag{2.30}$$

and the elements of $\mathbf{b}$ and $\mathbf{c}$ are (note that $\mathbf{c}$ is a constant vector):

$$b_{l_T} = -\frac{1}{2}\mu'_{l_T}\mathbf{\Sigma}^{-1}\mu_{l_T} \tag{2.31}$$

$$c_{l_T} = -\frac{K}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{\Sigma}| - \frac{1}{2}\mathbf{s}'\mathbf{\Sigma}^{-1}\mathbf{s} \tag{2.32}$$

Discriminative Gaussian Backend

In this case, ML estimates of the means and the common covariance matrix are used initially, but further reestimates of the means are iteratively computed in order to

maximize the Maximum Mutual Information (MMI) criterion:

$$F_{\mathbf{MMI}}(\lambda) = \sum_{\forall \mathbf{s}} \log \frac{p_\lambda(\mathbf{s}|l_T(\mathbf{s}))^C}{\sum_{\forall l} p_\lambda(\mathbf{s}|l)^C p(l)} \tag{2.33}$$

where $p_\lambda(\mathbf{s}|l(\mathbf{s}))$ is the likelihood of the score vector $\mathbf{s}$ given the true target language $l_T(\mathbf{s})$ and model parameters λ, $p(l)$ is the probability of language l and C is a heuristic factor.

Logistic Regression

Multiclass logistic regression [150] can be used to transform the scores following:

$$\hat{\mathbf{s}} = \mathbf{Q}\mathbf{s} + \mathbf{u} \tag{2.34}$$

where $\mathbf{Q}$ and $\mathbf{u}$ parameters are estimated to optimize the multi-class C_{LLR} (see Section 2.7).

2.6.3 Fusion

In the fusion stage, several systems can be combined. As outlined in Figure 2.7, fusion classifiers take several calibrated system outputs (one score per language and system) and combine them to give the final set of calibrated and fused scores.

Fusion classifiers use development data to evaluate the performance of each system and estimate the weight that will be given to each system output (represented as SP in the Figure 2.7). In this way, scores are weighted so that the systems performing better attain a higher relevance in the final decision score (CS), and are adjusted to fit the threshold. Binary logistic regression and multiclass logistic regression are techniques normally used for parameter tuning. Focal [61] is a useful and versatile toolkit widely used in the community for the fusion of several systems.

2.7 Evaluation Metrics

As in any other classification task, SLR performance metrics rely on combinations of different types of errors. As represented in Figure 2.8, spoken language verification systems produce two types of errors:

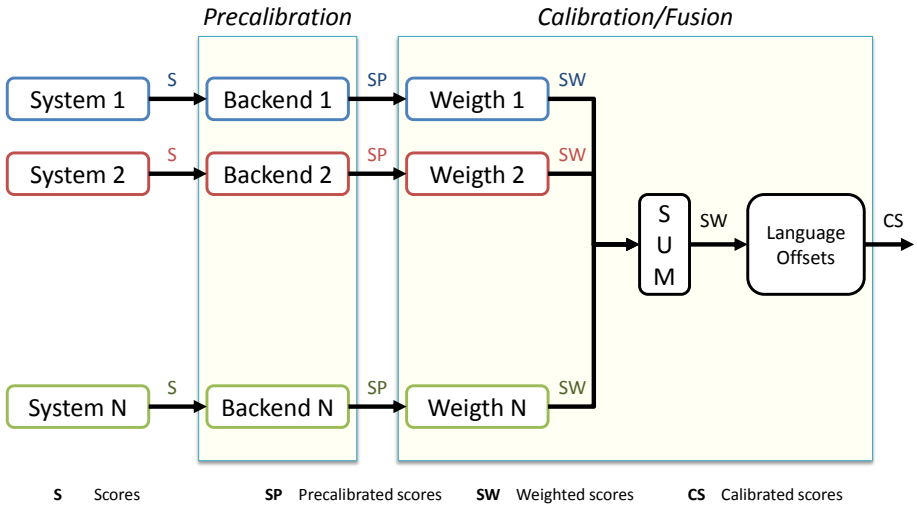


FIGURE 2.7: Calibration and fusion of several SLR systems

		True Label	
		Accept	***Reject***
Decision	***Accept***	Target trial	False alarm
	Reject	Miss	Non-target trial

FIGURE 2.8: Target and non-target trials and classification errors

- ***Misses:*** Trials for which the correct answer is *Accept* (target trials) but the system says *Reject*.
- ***False alarms:*** Trials for which the correct answer is *Reject* (non-target trials) but the system says *Accept*.

The corresponding error rates can be computed as:

- **Miss error rate,** P_{miss}**:** The fraction of target trials that are rejected.
- **False alarm error rate,** P_{fa}***:*** The fraction of non-target trials that are accepted.

Combinations of the error rates define different cost functions, in terms of which verification systems can be evaluated. The decision of the system may vary according to application dependent parameters of the cost functions: the prior probability of each language model (P_T), the cost of a miss error (C_{miss}) and the cost of a false alarm (C_{fa}), which define the operating point of the system. A well calibrated system should be able to automatically adapt the decisions to each particular application.

Equal Error Rate (EER)

This measure reports system performance at the operation point for which the false alarm error rate (P_{fa}) is equal to the miss error rate (P_{miss}). The EER does not measure the global performance of a system (i.e. for a wide range of operating points). Furthermore, since the threshold value is chosen *a posteriori* by the evaluator, it does not take into account the ability of the system to be positioned at the EER operation point (i.e. the performance loss due to bad calibration).

Detection Error Tradeoff (DET) curve

Detection Error Tradeoff (DET) curves [92] provide a straightforward way of comparing global performance of different systems for a given test condition and are used in NIST evaluations to support system performance comparisons.

A DET curve (as the one showed in Figure 2.9) is generated by computing P_{miss} and P_{fa} for a wide range of operation points (thresholds), based on the scores yielded by the analyzed system for a given test set. The axes of a DET curve show P_{fa} and P_{miss} in a lineal scale with regard to the normal distribution, given that both kinds of errors are assumed to follow that distribution. The scale is helpful to increment the resolution in low error regions, and makes lines representing systems look (usually) straight, making the comparison among systems easier. In each curve, two operating points are shown: the system operating point (given by system decisions, which depend on the chosen threshold) and the optimal (or minimum cost) operating point, which corresponds to the threshold that minimizes the cost function.

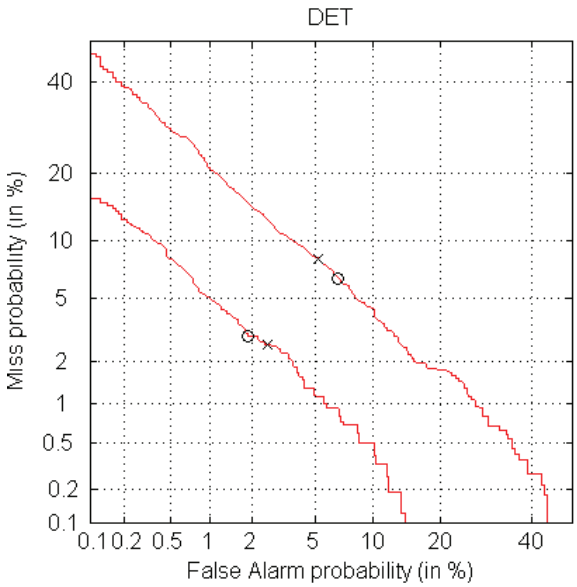


FIGURE 2.9: DET curves for two systems. The system operating point is marked with (x) whereas the optimal operating point is marked with (o).

Average Cost (C_{avg})

This measure is a combination of P_{miss} and P_{fa} and is computed as follows:

$$C_{\text{avg}} = \frac{1}{L}\sum_{i=1}^{L}\left\{\begin{array}{l} C_{\text{miss}}\cdot P_{\text{T}}\cdot P_{\text{miss(i)}} \\ +\frac{1}{L-1}\sum_{L_N} C_{\text{fa}}\cdot(1-P_{\text{T}}-P_{\text{oos}})\cdot P_{\text{fa}}\left(l_T, l_N\right) \\ +C_{\text{fa}}P_{\text{oos}}P_{\text{fa}}\left(l_T, l_O\right) \end{array}\right\} \tag{2.35}$$

where L is the number of target languages, l_T, l_N, and l_O stand for target, non-target and out-of-set languages, respectively; and the application-dependent parameters are: P_{T} the target prior, P_{oos} the prior for out of set languages (which takes the value 0 in closed set evaluations), C_{miss} the miss error cost and C_{fa} the false alarm error cost.

C_{avg} accounts for the calibration loss, but it is still limited to a single operation point. Since it was the primary measure in NIST evaluations until 2011, many authors have historically reported system performance in terms of C_{avg}, for this reason, though other measures will be also used to report SLR performance in this thesis, C_{avg} will be primarily used when commenting and comparing results.

NIST 2011 LRE metric, C_{avg}^{24}

In the NIST 2011 LRE, an alternative metric was used to evaluate system performance, based on a pairwise cost function defined by:

$$\begin{aligned} C(l_1, l_2) &= C_{\text{l}_1} P_{\text{l}_1} P_{\text{miss}}(l_1) \\ &\quad + C_{\text{l}_2}(1 - P_{\text{l}_1}) P_{\text{miss}}(l_2) \end{aligned} \tag{2.36}$$

where l_1 and l_2 denote languages 1 and 2, respectively.

The overall C_{avg}^{24} measure was defined as the mean of the $C(l_1, l_2)$ values over the 24 language pairs for which the C_{min} values were greatest[1]. In the evaluation, the application parameters were set to the following values: $C_{\text{l}_1} = C_{\text{l}_2} = 1$ and $P_{\text{l}_1} = 0.5$. For further details, see [8].

Log-Likelihood Ratio Cost (C_{LLR})

When the scores represent (or can be interpreted as) log-likelihoods, systems can be evaluated in terms of the so called C_{LLR} [20], which has been used as alternative performance measure in some NIST evaluations. C_{LLR} allows us to evaluate the system performance globally by means of a single numerical value. It only depends on the scores (it does not depend on application dependent parameters), on their ability to discriminate amongst target languages from each other and on how well they are calibrated, the two key features of a SLR system. On the other hand, it has higher statistical significance than EER or C_{avg}, since it is computed from verification scores (in contrast to EER or C_{avg}, which depend only on Accept/Reject decisions). Let us now recall how C_{LLR} is computed.

Let $LR(S, l_i)$ be the *likelihood ratio* corresponding to segment S and target language l_i. The likelihood ratio can be expressed in terms of the conditional probabilities of X with regard to the alternative target and non-target hypotheses, as follows:

$$LR(S, l_i) = \frac{p(S|l_i)}{p(S|\neg l_i)} \tag{2.37}$$

Let E be an evaluation dataset, consisting of the union of L disjoint subsets: E_{l_j} ($j \in [1, L]$) containing speech segments in the target language l_j. Pairwise costs

[1] $C_{\text{min}}(l_1, l_2)$: minimum of the pairwise cost function, found for the optimal operation point (threshold).

$C_{\text{LLR}}(l_i, l_j)$, for $i, j \in [1, L]$, are defined as follows:

$$C_{\text{LLR}}(l_i, l_j) = \begin{cases} \frac{1}{|E_{l_i}|} \sum\limits_{S \in E_{l_i}} \log_2(1 + LR(S, l_i)^{-1}) & j = i \\ \frac{1}{|E_{l_j}|} \sum\limits_{S \in E_{l_j}} \log_2(1 + LR(S, l_i)) & j \neq i \end{cases} \tag{2.38}$$

Finally, the average C_{LLR} is computed by adding the pairwise costs for all the combinations of target and non-target languages, as follows:

$$C_{\text{LLR}} = \frac{1}{L} \sum_{i=1}^{L} \{P_T \cdot C_{\text{LLR}}(l_i, l_i) + \sum_{\substack{j=1 \\ j \neq i}}^{L} P_N \cdot C_{\text{LLR}}(l_i, l_j)\} \tag{2.39}$$

where P_T is the prior probability of target languages and $P_N = (1 - P_T)/(L - 1)$ is the prior probability of non-target languages.

The C_{LLR} takes unbounded non-negative values expressed in information units (bits), with lower values representing better performance, the value 0 corresponding to a perfect system and the value $\log_2(L)$ corresponding to a system which just relies on priors, thus providing no information to decide a trial. C_{LLR} can be computed by means of the FoCal toolkit [61]. Further details about the reasons for using this measure and its interpretation can be found in [20] [22].

Actual Relative Confusion, F_{act}

In the Albayzin 2012 LRE, a new evaluation metric was proposed to measure system performance: F_{act}. This new metric measures the information provided by a SLR system through a set of log-likelihoods and does not require making hard decisions (see [124] for details).

To compute the metric, a prior distribution over language classes is specified, so that Bayes' rule can be used to map the submitted log-likelihoods to language class posteriors. The goodness of these posteriors is then evaluated by means of a logarithmic cost function. A weighted average of the logarithmic cost over all audio segments forms the cross-entropy criterion. In the following paragraphs the cross-entropy criterion is presented in the form of *relative confusion*, a measure closely related to perplexity.

Logarithmic cost function: for every audio segment, S_t, the system under evaluation submits the log-likelihood-vector, $\boldsymbol{\ell}_t$. The evaluator has access to the *true class label* for segment S_t, which we denote $l_{\text{true}(t)} \in \{l_1, \ldots, l_L\}$. This allows the

evaluator to compute a measure of goodness for $\boldsymbol{\ell}_t$, in the form of the *logarithmic cost function*:

$$C_{\log}(\mathbf{\Pi}_t | L_{\text{true}(t)}) = -\log P(l_{\text{true}(t)} | \boldsymbol{\ell}_t, \boldsymbol{\pi}) \tag{2.40}$$

where $\mathbf{\Pi}_t = \big(P(l_1|\boldsymbol{\ell}_t, \boldsymbol{\pi}), \ldots, P(l_L|\boldsymbol{\ell}_t, \boldsymbol{\pi})\big)$ is the whole posterior distribution, estimated according to the prior distribution $\boldsymbol{\pi}$ defined in [124] and using the softmax function to map the log-likelihood vector into posterior distributions.

Multiclass cross-entropy: the *evaluation criterion*, known as *multiclass cross-entropy*, is formed by a weighted average of the logarithmic cost:

$$\mathcal{C}_{\text{mce}} = \sum_{i=1}^{m} \frac{\pi_i}{\|\mathcal{T}_i\|} \sum_{t \in \mathcal{T}_i} -\log P(l_i | \boldsymbol{\ell}_t, \boldsymbol{\pi}) \tag{2.41}$$

where $\mathcal{T}_i$ is the subset of indices for segments of class i. By $\|\mathcal{T}_i\|$ we mean the number of segments of language class i.

The default system: the one that cannot make up its mind about the language class and outputs $\ell_{it} = k_t$ for every t. This gives $P(l_i|\boldsymbol{\ell}_t, \boldsymbol{\pi}) = \pi_i$ for every i, t.

$$\mathcal{C}_{\text{def}} = \sum_{i=1}^{L} -\pi_i \log \pi_i \tag{2.42}$$

which is just the prior entropy. If a submitted system has $\mathcal{C}_{\text{mce}} \geq \mathcal{C}_{\text{def}}$, then it does not improve upon the default system.

Confusion: to facilitate interpretation of cross-entropy, we define the *confusion* of the system under evaluation as:

$$F_{\text{mce}} = \exp(\mathcal{C}_{\text{mce}}) - 1 \tag{2.43}$$

Similarly, the *prior confusion* (confusion of the default system) is:

$$F_{\text{def}} = \exp(\mathcal{C}_{\text{def}}) - 1 \tag{2.44}$$

The *actual relative confusion* is defined as:

$$F_{\text{act}} = \frac{F_{\text{mce}}}{F_{\text{def}}} \tag{2.45}$$

The relative confusion is the factor by which the system has changed (hopefully reduced) the prior confusion. The reference value for relative confusion is 1. Badly calibrated systems that have relative confusion greater than one are doing worse

than the default system. Good systems must have relative confusion below 1. A perfect system would have relative confusion of zero.

Chapter 3

Phone Log-Likelihood Ratios

As exposed in Section 1, the main objective when searching for a new set of features was to find a way of conveying phonetic and spectral information into a single set of features.

In this chapter, the choice of features used in this work is presented and defined. Once the reader is familiar with the basic extraction procedure, details about the main approach in which the features are used are given.

After that, with the aim of optimizing the extraction of the new set of features, a detailed study is carried out using the NIST 2007 LRE dataset. This benchmark provides a good compromise between database size and reliability and generalization of the results, and was therefore selected as the primary benchmark for all the studies in this work.

Once the optimal configuration of the feature extraction procedure is found, the approach is compared with other acoustic and phonotactic approaches, to test the goodness of the representation. Besides, fusions at the score level are tried, involving all the mentioned approaches, to check the complementarity among different systems. The significance of the results is also studied. Finally, to check the robustness of the proposed features and to validate the conclusions attained on NIST 2007 LRE, three additional series of SLR experiments are presented and discussed on the following benchmarks: NIST 2009 LRE, NIST 2011 LRE and Albayzin 2010 LRE.

3.1 Definition of Phone Log-Likelihood Ratios (PLLR)

The computation of the features is based on a phone decoder that is assumed to provide a reasonable coverage of the phonetic content of most languages. Even if phoneme inventories in different languages range from 11 (e.g. for the East Papuan language Rotokas and the Amazonian language Pirahã) to 141 (e.g. for the African language !Xũ), the number of phonemes of most languages is between 30 and 60 phonetic units [37]. Let us consider one such phone decoder including N phone units, each of them represented typically by means of a model of S states. Given an input sequence of acoustic observations X, we assume that the acoustic posterior probability of each state s ($1 \leq s \leq S$) of each phone model i ($1 \leq i \leq N$) at each frame t, $p(i|s,t)$, is output by the phone decoder.

The number of units of the phone decoder, N, multiplied by the number of states that phone decoders provide for each phone unit would give a considerable feature vector size, intractable when combined with several post-processing steps usually applied on top of feature vectors. Therefore, to compute the features, first, the acoustic posterior probability for each phone unit i at each frame t is computed by adding the posteriors of its states:

$$p(i|t) = \sum_{\forall s} p(i|s,t) \tag{3.1}$$

This way, we obtain an N dimensional vector, the one that according to the phone decoder parameters, best describes the spectral content of the analysis window.

Geometrically, this vector can be seen as a point inside the *standard* $N-1$ *simplex*. The $N-1$ *simplex* is a subset of $\mathbb{R}^N$ determined by:

$$\Delta^{(n-1)} = \{(x_0, \ldots, x_{n-1}) \in \mathbb{R}^N \mid \sum_{i=0}^{n-1} x_i = 1 \ \wedge \ x_i \geq 0 \ \forall i\} \tag{3.2}$$

The vertices of this subset are given by the vectors $\mathbf{v}_i \in \mathbb{R}^N$, where:

$$\begin{aligned} v_1 &= (1,0,0,...,0,0) \\ v_2 &= (0,1,0,...,0,0) \\ &... \\ v_N &= (0,0,0,...,0,1) \end{aligned} \tag{3.3}$$

In our case, each vertex corresponds to a pure phonetic unit p_i.

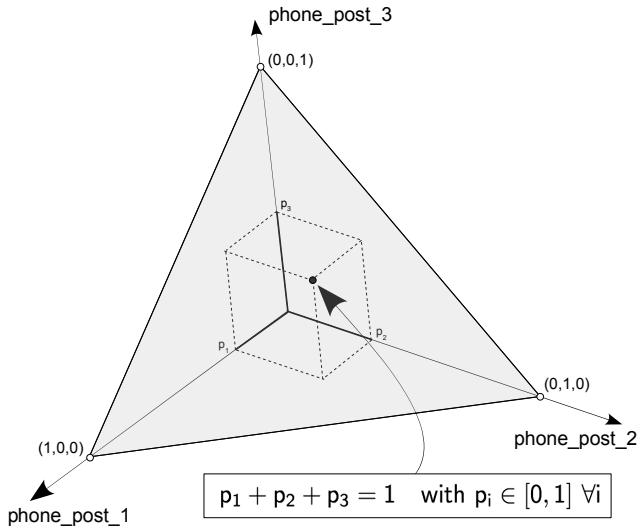


FIGURE 3.1: Standard 2-simplex defined by phone posteriors in the case of a phone decoder with 3 phonetic units.

Figure 3.1 shows the *Standard 2-simplex* for the case of a phone decoder with 3 phonetic units. In the graphic, each vertex represents one of the three pure phonetic units, and edges represent mixtures of the two phone units connected by them. According to the phone decoder, the closer the point represented by the feature vector is to each of the vertices, the closer the content of the analysis window is to that phone sound.

This vector already provides frame-by-frame information that could be used as a feature vector. The question at this point was, is it suitable as a feature vector? Most of the modeling approaches used in systems based on frame level features assume that they are normally distributed. When analyzing frame-level distributions of

phone posteriors (see Figure 3.2, row 1) they show really sparse behaviors, as it is expected for values representing probabilities. In order to compensate for this undesirable effect, we seek for ways of transforming the features. Among the vast number of possibilities, we selected the following, given the simplicity and result of the transformations: First, as shown in Figure 3.2, row 2, the frame-level log-posteriors are computed, whose distributions seem to be closer to Gaussian than those of posteriors, but still featuring some singularities. Then, after computing the phone posterior log-likelihood ratios (Figure 3.2, row 3), the distributions attained are seemingly Gaussian.

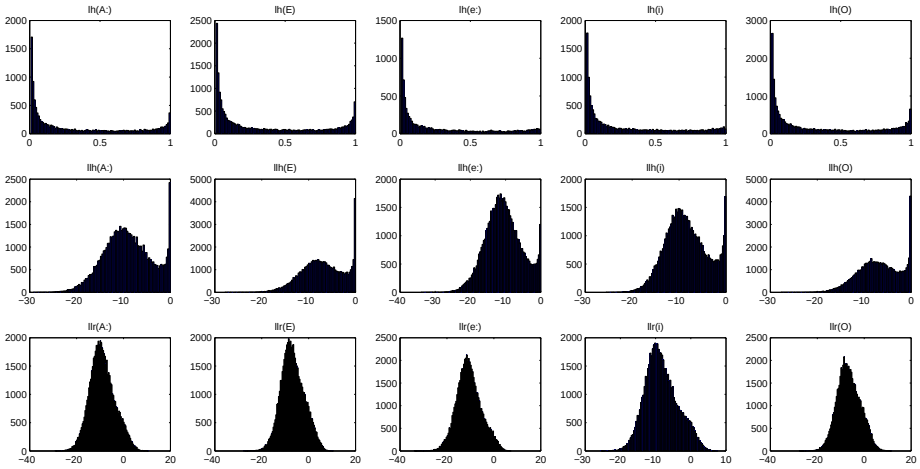


FIGURE 3.2: Distributions of frame-level phone posteriors (first row), phone log-posteriors (second row) and phone log-likelihood ratios (third row) for 5 phonetic units (A:, E, e:, i, O) of the Brno University of Technology decoder for Hungarian, computed on a subset of the NIST 2007 LRE test set.

Assuming a classification task with flat priors, phone log-likelihood ratios are computed from phone posterior probabilities as follows:

$$PLLR(i|t) = \log \frac{p(i|t)}{\frac{1}{(N-1)}(1 - p(i|t))} \qquad i = 1, ..., N \tag{3.4}$$

This way we obtain a feature vector carrying the same information as the feature vector output by the phone decoder, but featuring seemingly Gaussian distributions. We denote these features Phone Log-Likelihood Ratios (PLLRs).

3.2 Configuration of a SLR System Based on PLLR Features

In our approaches to compute the PLLRs, we use the open software Temporal Patterns Neural Network (TRAPs/NN) phone decoders, developed by the Brno University of Technology (BUT) [137]. Phone posterior probabilities are computed by adding the values corresponding to the three states of each phone unit (see eq. 3.1). BUT phone decoders provide posteriors for three non-phonetic units, which correspond to *int* (intermittent noise), *pau* (short pause) and *spk* (non-speech speaker noise). These three non-phonetic units were integrated into a single phone unit, by simply adding the values corresponding to their posteriors. Then, PLLRs were computed according to Equation 3.4, getting 43 (CZ), 59 (HU) and 50 (RU) log-likelihood ratios per frame, respectively, which we call PLLR features.

In [6] we provide open software to compute the PLLRs, either from phone posteriors, or from BUT phone decoder outputs.

Voice activity detection was performed by removing the feature vectors whose highest PLLR value corresponded to the non-phonetic unit.

The PLLR-based baseline system follows the *Total Variability Factor Analysis* (*i-vector*) approach (see section 2.4). A 1024-mixture gender-independent UBM with diagonal covariance matrix is trained with the maximum likelihood criterion, using binary mixture splitting, orphan mixture discarding and variance flooring.

A 500 dimensional total variability matrix is estimated as described in [50], using data from target languages. To model the spoken language, a generative model is defined in the i-vector feature space (as in [98]), the set of i-vectors of each language being represented by a single Gaussian.

3.3 Search for the Optimal PLLR Feature Configuration

Several configuration options could be tested on top of PLLRs. This section presents the studies carried out in order to contrast results of the systems trained with PLLR features modified or augmented with different techniques. Experimentation is carried out using the NIST 2007 LRE dataset (the one selected as development set for this work). In this study, among BUT TRAP/NN phone decoders, the one trained for Hungarian was selected for experimentation, given that it had provided good

results in several speech and language recognition tasks [43, 112, 143]. Results will be provided in terms of both C_{avg} and C_{LLR}, though performance will be mostly compared using C_{avg}, given that this is the primary measure used historically in most NIST LRE evaluations.

3.3.1 Phone Log-Likelihoods vs PLLRs

First, we compare the results attained by systems trained with PLLR features to those attained using phone log-posteriors (PL). Results in Table 3.1 clearly show that the PLLR transformation enhances the performance of the system.

TABLE 3.1: $C_{\text{avg}} \times 100$ and C_{LLR} performance for i-vector systems using phone log-posteriors (PL) features and PLLRs computed with the HU BUT decoder, on the NIST 2007 LRE primary evaluation task.

System	$C_{\text{avg}} \times 100$	C_{LLR}
PL	4.41	0.604
PLLR	3.45	0.564

3.3.2 Dynamic Coefficients

Dynamic coefficients have led to significant gains when applied on top of MFCC features in language verification systems, as they provide a larger temporal context, which augments the information carried out by the features. In our attempt to optimize the configuration of the PLLR-based SLR system, we tested the effect of computing dynamic coefficients on top of the PLLRs. In the experiments reported in this section, Deltas and double Deltas were estimated using equation 2.1 with values D=2 and D=1, respectively, and different systems were trained using PLLR, PLLR+Δ and PLLR+Δ+$\Delta\Delta$ feature sets. Results are shown in Table 3.2.

Using PLLR plus first order deltas yielded a 23% relative improvement in terms of C_{avg} with regard to using only PLLRs. Second order deltas, instead, degraded the performance of the system. This degradation could possibly be due to the high dimensionality of the feature set. When computing double Deltas with the HU BUT TRAP/NN phone decoder, the feature vector reaches dimensionality $59 \times 3 = 177$, which combined with the 1024 dimensional GMM we use in our system, gives a 181248 dimensional supervector. This could pose a problem, as the training data might not be enough for properly estimating all the parameters.

TABLE 3.2: $C_{avg} \times 100$ and C_{LLR} performance for i-vector systems using PLLR, PLLR+Δ and PLLR+Δ+ΔΔ features computed with the HU BUT decoder, on the NIST 2007 LRE primary evaluation task.

System	$C_{avg} \times 100$	C_{LLR}
PLLR	3.45	0.564
PLLR+Δ	2.66	0.382
PLLR+Δ+ΔΔ	3.60	0.506

In the experiments reported hereafter in this thesis, PLLR+Δ features will be used as baseline.

3.3.3 Variability Compensation

Several channel compensation techniques can be applied at the feature extraction stage (see Section 2.4). Among them, Feature Normalization and Feature Warping are two of the most extensively applied on features for language (and speaker) verification systems. Table 3.3 presents results for the baseline systems, and systems based on PLLR+Feature Normalization (FN), PLLR+Feature Warping (FW) and PLLR+RASTA.

TABLE 3.3: $C_{avg} \times 100$ and C_{LLR} performance for i-vector systems using PLLR features computed with the BUT HU decoder, with: (a) no noise reduction technique, (b) Feature Normalization (FN), (c) Feature Warping (FW) and (d) RASTA, on the NIST 2007 LRE primary evaluation task.

System	$C_{avg} \times 100$	C_{LLR}
PLLR	2.66	0.382
PLLR+FN	2.95	0.436
PLLR+FW	3.21	0.435
PLLR+RASTA	8.67	1.149

Figures show that these techniques provide no gain with regard to the basic PLLR feature-based system, thus it was decided not to apply any variability compensation technique on top of the PLLRs at the feature extraction stage.

3.4 Overall Performance of PLLR Based Systems

Once the optimal configuration was selected, that is, PLLR features augmented with first order deltas and with no variability compensation techniques applied at the feature extraction stage, we studied the performance of the systems trained on the features obtained with different decoders and on different benchmarks.

First, results are presented for the NIST 2007 LRE dataset using the BUT TRAP/NN CZ, HU and RU phone decoders. Results are also presented for different state-of-the-art systems: an acoustic system, which shares the modeling part with the PLLR approach, and three phone-lattice-SVM systems, which use the same phone decoders as the ones used for the PLLR approaches, thus sharing the origin of the features. System fusions have been also explored to check the complementarity between approaches. Finally, results are also presented for the NIST 2009 LRE, NIST 2011 LRE and Albayzin 2010 LRE datasets.

Details about system configuration, the applied backends and fusion procedures are given below:

Baseline MFCC-SDC i-vector System

The concatenation of MFCC and SDC coefficients under a 7-2-3-7 configuration was used as acoustic representation for the baseline acoustic i-vector system (like in previous works [109] [114]). Voice activity detection was performed by removing the feature vectors whose highest PLLR value corresponded to the non-phonetic unit using the Brno University of Technology decoder for Hungarian (see Section 2.2.2 for details).

The GMM configuration, the estimation of the total variability matrix and scoring were also performed as for the PLLR i-vector system, that is: A gender independent 1024-mixture UBM was estimated by the Maximum Likelihood criterion on the training dataset, using binary mixture splitting, orphan mixture discarding and variance flooring. The total variability matrix T was estimated according to the procedure defined in [40], using only data from target languages. A generative modeling approach was applied in the i-vector feature space, the set of i-vectors of each language being modeled by a single Gaussian distribution.

Baseline Phonotactic Systems

The three phonotactic systems applied in this work were developed under the Phone-lattice-SVM approach (see Section 2.3). Given an input signal, an energy-based voice activity detector was applied. Then, the open software BUT Temporal Patterns Neural Network (TRAPs/NN) CZ, HU and RU phone decoders were applied.

Regarding channel compensation, noise reduction, etc. the three systems relied on the acoustic front-end provided by BUT decoders.

BUT decoders were configured to produce phone posteriors that were converted to phone lattices by means of HTK [156] along with the BUT recipe [137]. Then, expected counts of phone n-grams were computed using the *lattice-tool* of SRILM [144]. Finally, an SVM classifier was applied, SVM vectors consisting of expected frequencies of phone n-grams (up to $n = 3$), weighted as in [121]. A sparse representation was used, which involved only the most frequent features according to a greedy feature selection algorithm [111]. L2-regularized L1-loss support vector regression was applied, by means of LIBLINEAR [59].

Backend and Fusion Models

In this work, the backend setup was optimized in preliminary experiments on the development set of each database, and then applied to the corresponding evaluation set (see Appendix A for dataset configuration details). For the NIST 2007 and 2009 LRE datasets, a ZT-norm and a discriminative Gaussian backend were applied to scores. For the NIST 2011 LRE dataset a generative Gaussian backend was applied. Raw system scores were used for he Albayzin 2010 dataset.

In all cases, backend parameters were estimated and applied by means of the *FoCal* toolkit [20] [22] [17] [62]. When combining systems, discriminative logistic regression fusion parameters were estimated also using *FoCal*.

3.4.1 Results on the NIST 2007 LRE dataset

Table 3.4 presents results for this dataset. Regarding individual system performances, the acoustic system reaches 2.85 C_{avg}. Performance of phonotactic systems ranges from 2.08 C_{avg} for the one using the HU phone decoder, to 2.94 C_{avg} for the one using the CZ phone decoder. The same happens with the PLLR systems, the best results are attained with the features based on the HU phone decoder, 2.66 C_{avg}, and performance degrades up to 4.18 C_{avg} when using the CZ decoder. Analyzing the results, we see that the best PLLR result is in between the performance of the best phone-lattice-SVM system and the one attained by the acoustic system.

Looking at results of fusions, as expected, acoustic and phonotactic approaches combine well, HU-Phonotactic + acoustic-i-vector reaches 1.08 C_{avg}. Acoustic and PLLR based i-vector systems, though sharing all the modeling part, also combine well, attaining 1.40 C_{avg} (when combined with the HU decoder based PLLRs). The result of fusing phonotactic and PLLR based approaches, even though they share the origin of the features, provides a significant gain, getting up to 1.20 C_{avg}. Most

TABLE 3.4: $C_{\text{avg}} \times 100$ and C_{LLR} performance for the MFCC-SDC i-vector baseline system, i-vector systems using PLLR features, phonotactic baseline systems and the fusion of them, for each of the BUT decoders, on the NIST 2007 LRE primary evaluation task.

System			$C_{\text{avg}} \times 100$	C_{LLR}
MFCC-SDC i-vector		(a)	2.85	0.407
CZ	Phonotactic	(b1)	2.94	0.440
	PLLR i-vector	(c1)	4.18	0.550
Fusion	(a)+(b1)		1.22	0.189
	(a)+(c1)		1.95	0.280
	(b1)+(c1)		1.79	0.257
	(a)+(b1)+(c1)		1.24	0.176
HU	**Phonotactic**	**(b2)**	**2.08**	**0.310**
	PLLR i-vector	(c2)	2.66	0.382
Fusion	(a)+(b2)		1.08	0.152
	(a)+(c2)		1.40	0.215
	(b2)+(c2)		1.20	0.166
	(a)+(b2)+(c2)		**0.82**	**0.124**
RU	Phonotactic	(b3)	2.69	0.383
	PLLR i-vector	(c3)	4.08	0.549
Fusion	(a)+(b3)		1.13	0.182
	(a)+(c3)		1.72	0.265
	(b3)+(c3)		1.76	0.240
	(a)+(b3)+(c3)		1.10	0.163

remarkably, when fusing the three approaches, the system gets 0.82 C_{avg}, meaning that PLLR features provide complementary information with regard to both, acoustic and phonotactic approaches. These results are consistent also when using other decoders, gains being more remarkable when comparing C_{LLR} metrics.

Statistical Significance

To measure the statistical significance of performance improvements (in terms of C_{avg}, which is the primary performance measure in this work), a series of two-tailed paired T-tests was carried out [68], which gives an idea of the variability of performance improvements (and thus, the robustness of such improvements) across randomly defined sets of data. To that end, the NIST LRE 2007 evaluation dataset was split into 20 language-balanced disjoint random subsets. Then, C_{avg} values were computed on each subset for baseline systems (a), (b2) and (a)+(b2) and for the same

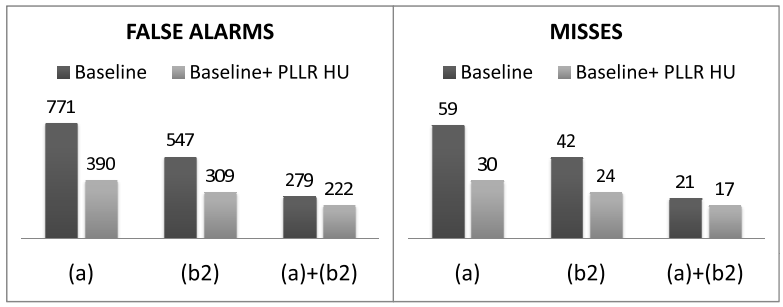


FIGURE 3.3: False alarm and miss errors on the NIST LRE 2007 primary task for baseline systems: (a) acoustic MFCC-SDC i-vector system, (b2) Phone-Lattice-SVM system and (a)+(b2) the fusion of the two latter, taken alone (dark gray) and fused with (c2) the HU PLLR i-vector system (light gray).

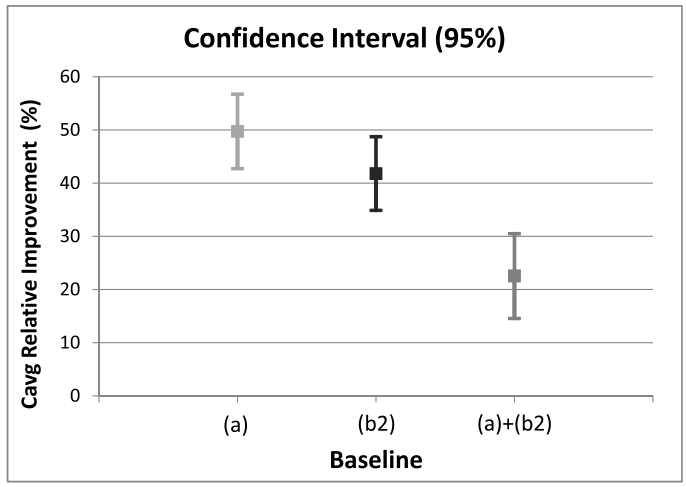


FIGURE 3.4: Means of C_{avg} relative improvements and their corresponding intervals at 95% confidence level, on the NIST LRE 2007 primary task, when fusing the HU PLLR i-vector system (c2) with baseline systems: (a) acoustic MFCC-SDC i-vector system, (b2) Phone-Lattice-SVM system and (a)+(b2) the fusion of the two latter.

systems fused with the PLLR HU system: (a)+(c2), (b2)+(c2) and (a)+(b2)+(c2). Figure 3.3 provides the number of false alarm and misses for the systems. Figure 3.4 shows the mean and the confidence interval at 95% confidence level of the relative C_{avg} improvements, revealing that they are statistically significant in all cases.

3.4.2 Results on the NIST 2009 LRE dataset

When analyzing the results attained in this benchmark, presented in Table 3.5, we find that the best single system is the Phonotactic-RU, yielding 2.24 C_{avg}, followed by the PLLR i-vector HU phonotactic system with 2.42 C_{avg}, and finally the MFCC-SDC i-vector, which obtains 2.70 C_{avg}.

TABLE 3.5: $C_{\text{avg}} \times 100$ and C_{LLR} performance for the baseline phonotactic and i-vector systems, the PLLR i-vector system and the fusion of them, on the NIST 2009 LRE primary evaluation task.

System		$C_{\text{avg}} \times 100$	C_{LLR}
MFCC-SDC i-vector (a)		2.70	0.535
CZ	Phonotactic (b1)	2.98	0.578
	PLLR i-vector (c1)	3.18	0.592
Fusion	(a)+(b1)	1.82	0.365
	(a)+(c1)	1.99	0.416
	(b1)+(c1)	1.95	0.399
	(a)+(b1)+(c1)	1.61	0.388
HU	Phonotactic (b2)	2.49	0.502
	PLLR i-vector (c2)	**2.42**	**0.505**
Fusion	(a)+(b2)	1.67	0.346
	(a)+(c2)	1.79	0.392
	(b2)+(c2)	1.69	0.357
	(a)+(b2)+(c2)	**1.48**	**0.321**
RU	Phonotactic (b3)	2.24	0.457
	PLLR i-vector (c3)	2.74	0.548
Fusion	(a)+(b3)	1.53	0.323
	(a)+(c3)	1.92	0.404
	(b3)+(c3)	1.65	0.344
	(a)+(b3)+(c3)	1.47	0.307
Fusion	Phonotactics (b1+b2+b3)	1.66	0.343
	PLLRs (c1+c2+c3)	1.89	0.402
	(a)+(b1+b2+b3)	1.43	0.295
	(a)+(c1+c2+c3)	1.69	0.361
	ALL	**1.28**	**0.282**

Regarding fusions, once again the acoustic and phonotactic approaches combine well, followed closely by the fusion of the phonotactic and PLLR i-vector systems. The fusion of the two i-vector systems (acoustic and PLLR-based) yields similar figures.

The fusion of the acoustic system with any pair of phonotactic and PLLR i-vector systems leads to improved performance in all cases. Other fusions are also presented in Table 3.5, remarkably the fusion of the three phonotactic systems, which achieves better performance than the fusion of the PLLR-based systems. The fusion of all 7 systems attains a remarkable 1.28 C_{avg}.

3.4.3 Results on the NIST 2011 LRE dataset

Performance on the NIST 2011 LRE dataset is presented in Tables 3.6 (old C_{avg} metric) and 3.7 (new NIST 2011 C_{avg}^{24} metric).

In this benchmark, the best single system is PLLR-RU, which reaches 4.70 C_{avg}. The MFCC-SDC i-vector system obtains 5.96 C_{avg} and the best phonotactic system gets 6.85 C_{avg}. Best pairwise fusions are obtained when combining both i-vector approaches, reaching 3.77 C_{avg}. The fusion of the three systems yields up to 3.37 C_{avg}. In NIST 2011 LRE the fusion of the three PLLR-based systems clearly outperforms the fusion of phonotactic systems. The overall fusion reaches 3.01 C_{avg}.

Results measured with the new metric are consistent with the ones obtained using C_{avg}, but with the new metric differences among systems are more noticeable, and relative improvements when fusing systems are (overall) more significant.

TABLE 3.6: $C_{avg} \times 100$ and C_{LLR} performance for the i-vector baseline system using acoustic features (MFCC-SDC), i-vector systems with PLLR+Δ features, phonotactic baseline systems and the fusion of them, for each of the BUT decoders, on the NIST 2011 LRE primary evaluation task.

System		$C_{avg} \times 100$	C_{LLR}
MFCC-SDC i-vector (a)		5.96	1.088
CZ	Phonotactic (b1)	7.98	1.376
	PLLR+Δ i-vector (c1)	5.31	0.978
Fusion	(a)+(b1)	4.34	0.816
	(a)+(c1)	4.01	0.770
	(b1)+(c1)	4.36	0.822
	(a)+(b1)+(c1)	3.64	0.691
HU	Phonotactic (b2)	7.15	1.280
	PLLR+Δ i-vector (c2)	5.18	0.982
Fusions	(a)+(b2)	4.34	0.823
	(a)+(c2)	4.00	0.789
	(b2)+(c2)	4.39	0.829
	(a)+(b2)+(c2)	3.63	0.714
RU	Phonotactic (b3)	6.85	1.212
	PLLR+Δ i-vector (c3)	**4.70**	**0.898**
Fusions	(a)+(b3)	4.25	0.780
	(a)+(c3)	3.77	0.734
	(b3)+(c3)	**3.91**	**0.740**
	(a)+(b3)+(c3)	**3.37**	**0.647**
Fusions	Phonotactics (b1)+(b2)+(b3)	4.57	0.844
	PLLRs (c1)+(c2)+(c3)	**3.79**	**0.720**
	(b1+b2+b3)+(c1+c2+c3)	3.13	0.607
	(a)+(b1+b2+b3)	3.58	0.674
	(a)+(c1+c2+c3)	3.39	0.663
	ALL	**3.01**	**0.578**

TABLE 3.7: min $C_{avg}^{24} \times 100$ and actual $C_{avg}^{24} \times 100$ performance for the phonotactic and acoustic i-vector baseline systems, the PLLR+Δ i-vector system and the fusion of them, on the NIST 2011 LRE primary evaluation task.

System		min $C_{avg}^{24} \times 100$	$C_{avg}^{24} \times 100$
MFCC-SDC i-vector (c)		11.63	13.56
CZ	Phonotactic (a1)	13.59	15.05
	PLLR+Δ i-vector (b1)	10.00	12.46
Fusion	(a)+(b1)	8.32	10.41
	(a)+(c1)	7.36	10.04
	(b1)+(c1)	7.95	10.00
	(a)+(b1)+(c1)	6.59	8.85
HU	Phonotactic (a2)	12.49	14.28
	PLLR+Δ i-vector (b2)	9.83	12.12
Fusions	(a)+(b2)	7.98	0.10.43
	(a)+(c2)	7.78	10.19
	(b2)+(c2)	7.75	10.31
	(a)+(b2)+(c2)	6.68	9.14
RU	Phonotactic (a3)	11.37	12.91
	PLLR+Δ i-vector (b3)	**8.27**	**11.27**
Fusions	(a)+(b3)	7.59	9.38
	(a)+(c3)	6.73	9.21
	(b3)+(c3)	**7.36**	**10.04**
	(a)+(b3)+(c3)	**5.61**	**7.96**
Fusions	Phonotactics (b1)+(b2)+(b3)	7.86	9.62
	PLLRs (c1)+(c2)+(c3)	**6.57**	**9.10**
	(b1+b2+b3)+(c1+c2+c3)	5.06	7.51
	(a)+(b1+b2+b3)	5.99	8.47
	(a)+(c1+c2+c3)	5.88	8.80
	ALL	**4.78**	**7.49**

3.4.4 Results on the Albayzin 2010 LRE dataset

With the aim of exploring possible performance differences when dealing with *wide-band* signals, and specially in noisy environments, SLR experiments were also carried out on the Albayzin 2010 LRE dataset.

TABLE 3.8: $C_{\text{avg}} \times 100$ and C_{LLR} performance for the baseline systems, and the PLLR i-vector systems using phone decoders for CZ, HU and RU on the Albayzin 2010 LRE primary task on clean and noisy speech.

System		Clean		Noisy	
		$C_{avg} \times 100$	C_{LLR}	$C_{avg} \times 100$	C_{LLR}
MFCC-SDC i-vector		2.12	0.176	3.95	0.325
CZ	Phonotactic	2.15	0.215	7.00	0.664
	PLLR i-vector	2.33	0.223	6.66	0.546
HU	Phonotactic	2.35	0.218	7.28	0.621
	PLLR i-vector	**1.41**	**0.127**	**3.17**	**0.308**
RU	Phonotactic	2.85	0.244	6.54	0.571
	PLLR i-vector	2.34	0.225	4.38	0.352

Results are consistent with the ones obtained in the NIST benchmarks (remind that all of them involved 8 kHz telephone-channel speech). Focusing first on clean speech results (see Table 3.8), the MFCC-SDC i-vector system reaches 2.12 in terms of C_{avg}. Comparing the results attained by the systems trained with different decoders, we see that this time the best phonotactic system is the one trained with the CZ phone decoder (2.15 C_{avg}), followed closely by the HU one (2.35 C_{avg}). Among the PLLR i-vector approaches, instead, the HU attains a remarkable 1.41 C_{avg}, making it the best individual system. Performance on the noisy condition suffers (obviously) a severe degradation, more pronounced in phonotactic systems. The acoustic i-vector system obtains 3.95 C_{avg}. Among phonotactic systems, the one trained with the RU decoder provides the best performance (6.54 C_{avg}). On the other hand, the HU i-vector PLLR system outperforms all the individual systems also in noisy speech, attaining 3.17 C_{avg}.

Table 3.9 shows the performance attained by the baseline MFCC-SDC i-vector system, the baseline phonotactic-HU system, the PLLR-HU i-vector system and fusions of them. As in other benchmarks, all pairwise system combinations obtain good results. In clean speech, the fusion of the MFCC-SDC i-vector system and phonotactic-HU led to great improvements with regard to single system performance (1.10 C_{avg}). Similarly, the combination of the phonotactic-HU and PLLR-HU systems attained 1.09 C_{avg}, followed closely by the fusion of the two i-vector systems (1.20 C_{avg}).

TABLE 3.9: $C_{\text{avg}} \times 100$ and C_{LLR} performance for the baseline systems, the PLLR i-vector system and different fusions on the Albayzin 2010 LRE primary task on clean and noisy speech.

System			Clean		Noisy	
			$C_{avg} \times 100$	C_{LLR}	$C_{avg} \times 100$	C_{LLR}
MFCC-SDC i-vector		(a)	2.12	0.176	3.95	0.325
HU	Phonotactic	(b)	2.35	0.218	7.28	0.621
	PLLR i-vector	**(c)**	**1.41**	**0.127**	**3.17**	**0.308**
Fusion	(a)+(b)		1.10	0.106	2.43	0.211
	(a)+(c)		1.20	0.109	2.65	0.227
	(b)+(c)		1.09	0.092	2.65	0.228
	(a)+(b)+(c)		**0.97**	**0.086**	**1.86**	**0.168**
Fusion	ALL (7 systems, Table 3.8)		0.82	0.075	1.74	0.169

The fusion of the three systems still provided some further improvements, leading to 0.97 C_{avg}, which means a 12% relative improvement with regard to the best pairwise fusion. In the noisy condition, all pairwise fusions obtain similar performance. The best out of them is the combination of MFCC-SDC i-vector and phonotactic systems (2.43 C_{avg}), followed by the other two, which obtain the same figure (2.65 C_{avg}). The fusion of the three systems reaches 1.86 C_{avg}, providing a 23% relative improvement with regard to the best pairwise fusion.

In this benchmark, the fusion of all 7 systems still provides a slight improvement in the overall performance.

3.5 Chapter Summary

In this chapter, we have defined the PLLR features, integrated them in a language recognition system, presented an study focusing on the optimization of a PLLR-based system and evaluated its performance in several benchmarks.

The studies have revealed that PLLR features are easy to extract and integrate in state-of-the-art systems. Their usefulness has been extensively validated, as systems based on PLLR features attain competitive results in the four analyzed benchmarks.

Furthermore, the high performance attained by fusions of the PLLR-based system with baseline acoustic and phonotactic approaches reveals a complementarity with state of the art techniques, and the suitability of using PLLR features to improve overall system performance.

Chapter 4

Dimensionality Reduction on PLLRs

As exposed in the previous chapter, PLLRs are an effective way of conveying acoustic and phonetic information into frame level vectors. This, along with the fact that they can be computed using open-software, makes them easy to integrate in other state-of-the-art approaches based on frame level representations. Nevertheless, PLLR features have higher dimensionality than the acoustic representations that they are *replacing* in the systems. This can pose a computational problem when trying to deal with certain post-processing or modeling approaches.

MFCC feature vectors usually range from 7 to 19 dimensions, which are then augmented with Dynamic coefficients or with SDC, which doubles, triples or even multiplies by 7 the size of the feature vectors [139], [18]. The size of PLLR feature vectors depends on the phone decoder and the phonetic inventory of the language it's trained on. As outlined in Section 3.1, though most languages have around 30 phonemes, the number of units of the phone decoders used in this work ranges from 30 to 60 (see Section 3.2 for details).

This Chapter deals with the high dimensionality issue, by studying different reduction techniques that can be applied to PLLR features.

4.1 Supervised and Unsupervised Dimensionality Reduction Techniques

There is a handful of works in different fields of speech recognition literature, aiming to reduce the number of phone models into smaller broad classes, either by clustering or by selection techniques. On the one hand, some works apply supervised clustering, which requires knowledge of the language/phonemes to define phone families, as in [79], where several phone sets are defined for a PRLM approach used in a SLR task, or in [143], where a reduced phone set is also used to reduce n-gram counts on a phonotactic SLR i-vector approach. On the other hand, unsupervised clustering based on different distance metrics like the confusion among phonemes [157] or mutual information based merging and selection [88] have also been applied to improve speech or language recognition.

4.1.1 Supervised Techniques

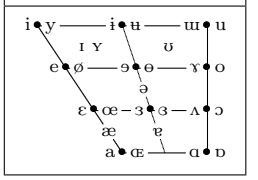
In phonotactic SLR approaches, it is a common practice to take advantage of the phonetic knowledge to reduce the set of phone units [79], [143]. Different clusterings can be performed in the phone posterior probability space (on which PLLRs are computed) based on expert knowledge of the properties that make each phoneme different from others (manner and place of articulation, voicing, etc.).

The International Phonetic Alphabet (IPA) [9] is a phonetic notation system built by linguists in order to provide a normalized and unique way of representing all the possible sounds of spoken languages. It contains 107 symbols and 55 modifiers. These symbols are mostly based on the Latin alphabet (using as few non-Latin forms as possible) and they are classified in different categories: letters represent basic sounds, that is *pulmonic consonants*, *non-pulmonic consonants* and *vowels*; *diacritics* specify these sounds, *suprasegmentals* point out special qualities of the sounds such as stress or durations, *tones and word accents* are specified by their level and contour and the rest of possible variations are covered by *other symbols* (see Figure 4.1 for details).

PULMONIC CONSONANTS											
	Bilabial	Lab. dent.	Dental	Alveolar	P-alveo.	Retroflex	Palatal	Velar	Uvular	Pharyng.	Glottal
Plosive	p b		t d			ʈ ɖ	c ɟ	k g	q ɢ		ʔ
Nasal	m	ɱ	n			ɳ	ɲ	ŋ	ɴ		
Trill	ʙ		r						ʀ		
Tap/Flap			ɾ			ɽ					
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ	ħ ʕ	h ɦ
Lat.Fric.			ɬ ɮ								
Approx		ʋ	ɹ			ɻ	j	ɰ			
Lat.appr.			l			ɭ	ʎ	ʟ			

NON PULMONIC CONSONANTS					
Clicks		Voiced implosives		Ejectives	
ʘ	Bilabial	ɓ	Bilabial	’	Examples:
|	Dental	ɗ	Dental/alveolar	pʼ	Bilabial
!	(Post)alveolar	ʄ	Palatal	tʼ	Dental/alveolar
ǂ	Palatoalveolar	ɠ	Velar	kʼ	Velar
||	Alveolar lateral	ʛ	Uvular	sʼ	Alveolar fricative

VOWELS



DIACRITICS								
◌̥	Voiceless	n̥	◌̤	Breathy voiced	b̤	◌̪	Dental	t̪
◌̬	Voiced	s̬	◌̰	Creaky voiced	b̰	◌̺	Apical	t̺
◌ʰ	Aspirated	tʰ	◌̼	Linguolabial	t̼	◌̻	Laminal	t̻
◌̹	More rounded	ɔ̹	◌ʷ	Labialized	tʷ	◌̃	Nasalized	ẽ
◌̜	Less rounded	ɔ̜	◌ʲ	Palatalized	tʲ	◌ⁿ	Nasal release	dⁿ
◌̟	Advanced	u̟	◌ˠ	Velarized	tˠ	◌ˡ	Lateral release	dˡ
◌̠	Retracted	e̠	◌ˤ	Pharyngealized	tˤ	◌̚	No audible release	d̚
◌̈	Centralized	ë	~	Velar. or pharyng.	ɬ			
◌̽	Mid-Centralized	ě	◌̝	Raised	e̝	(ɹ̝=voiced alveolar fricative)		
◌̩	Syllabic	n̩	◌̞	Lowered	e̞	(β̞=voiced bilabial approximant)		
◌̯	Non-syllabic	e̯	◌̘	Advanced Tongue Root	e̘			
◌˞	Rhoticity	ɚ	◌̙	Retracted Tongue Root	e̙			

TONES AND WORD ACCENTS			
é̋ , ˥e	Extra high	ě , ˧˥e	Rising
é , ˦e	High	ê , ˥˧e	Falling
ē , ˧e	Mid	˦˥e	High rising
è , ˨˧v	Low	˧˩v	Low rising
ȅ , ˩v	Extra low	˦˥˧v	(High) rising falling

SUPRASEGMENTALS	
ˈ	Primary stress
ˌ	Secondary stress
eː	Long
eˑ	Half-long
ĕ	Extra short
|	Minor (foot) group
||	Major (intonation) group
.	Syllable break
‿	Linking (absence of a break)

OTHER SYMBOLS			
ʍ	Voiceless labial-velar fricative	ɕ ʑ	Alveolo-palatal fricatives
w	Voiced labial-velar approximant	ɺ	Voiced alveolar lateral flap
ɥ	Voiced labial-palatal approximant	ʃʃ	Simultaneous ʃ and x
ɧ	Voiceless epiglottal fricative	Affricates and double articulations can be represented by tow symbols joined by a tie bar if necessary	k͡p t͡s
ʢ	Voiced epiglottal fricative		
ʔ	Epiglottal plosive		

FIGURE 4.1: IPA charts

In our study, we used these tables as reference, and applied them to classify phonetic units in the BUT TRAP/NN HU phone decoder. The Hungarian language consists of the set of phonemes shown in Table 4.1 (plus long forms of some of them, denoted by the suprasegmental ":") and Figure 4.2.

TABLE 4.1: IPA chart for the consonant phonemes of Hungarian

	Bilabial	Lab. dent.	Dental	Alveolar	P-alveo.	Palatal	Velar	Glottal
Plosive	p b			t d		c ɟ	k g	
Nasal	m			n		ɲ		
Trill				r				
Fricative		f v		s z	ʃ ʒ			h
Approximant						j		
Lat. approximant				l				
Affricate				t͡s d͡z	t͡ʃ d͡ʒ	c͡ç ɟ͡ʝ*		

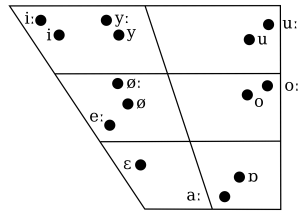


FIGURE 4.2: IPA chart for vowel phonemes of Hungarian

Four different phone classifications were considered in this study:

- *Family-R*: The set of Reduced (R) phones used in [143][1] to reduce the number of n-gram counts in a phonotactic approach, resulting in 33 phone classes.
- *Family-SL*: A set of phonemes defined by merging all Short and Long (SL) phonemes (for m, n, b, t, s, z, r, l, ʃ, c, ɟ, ɲ, j and k), the allophones (for m, n and h), and vowels according to their basic sound, resulting in 31 phone classes.
- *Family-MP*: A set of phonemes defined according to phonetic categories following IPA charts. Phones produced with the same Manner and Place (MP) of articulation were merged. Vowels belonging to the same regions of the IPA chart were also merged. This merging provided 23 phone classes. Table 4.2 shows by different colorings the clusters defined for consonant phonemes in this approach.
- *Family-M*: A more generic phonetic classification, where consonants produced with the same Manner (M) of articulation were merged. Vowels belonging

[1] The set of merged phonemes was kindly provided by the authors.

TABLE 4.2: IPA chart for the consonant phonemes of Hungarian merged according to Family-MP criteria

	Bilabial	Lab. dent.	Dental	Alveolar	P-alveo.	Palatal	Velar	Glottal
Plosive	p b		t d			c ɟ	k g	
Nasal	m		n			ɲ		
Trill			r					
Fricative		f v		s z	ʃ ʒ			h
Approximant						j		
Lat. approximant			l					
Affricate				t͡s d͡z	t͡ʃ d͡ʒ	c͡ç ɟ͡ʝ*		

to the same regions in the IPA charts were also merged, attaining 14 phone classes. Table 4.3 shows by different colorings the clusters defined for consonant phonemes in this approach.

TABLE 4.3: IPA chart for the consonant phonemes of Hungarian merged according to Family-M criteria

	Bilabial	Lab. dent.	Dental	Alveolar	P-alveo.	Palatal	Velar	Glottal
Plosive	p b		t d			c ɟ	k g	
Nasal	m		n			ɲ		
Trill			r					
Fricative		f v		s z	ʃ ʒ			h
Approximant						j		
Lat. approximant			l					
Affricate				t͡s d͡z	t͡ʃ d͡ʒ	c͡ç ɟ͡ʝ*		

For each of the above families, phones included in the same phonetic class were used to define a single unit by adding the posteriors obtained in Equation 3.1, before computing the log-likelihood ratios.

4.1.1.1 Results and Selection of the Optimal Supervised Technique

Table 4.4 shows performance of the different supervised dimensionality reduction techniques, compared to the baseline system (based on the PLLR+Δ features presented on Chapter 3[2]).

The system trained on the baseline features (with a PLLR feature vector of size 59 plus Deltas) attains 2.86 C_{avg}. When using the 33 dimensional *Family-R* feature set (decreasing the feature vector size to almost a half), performance is just slightly degraded, obtaining 3.07 C_{avg}. The system trained on the *Family-SL* feature set

[2]Note that baseline performance reported in this Chapter is worse than that reported in Chapter 3, due to the lower number of iterations performed to obtain the Total Variability matrix (5 instead of 10).

TABLE 4.4: $\%C_{\text{avg}}$ and C_{LLR} performance for the PLLR i-vector system with different knowledge-based phone merging approaches, on the NIST 2007 LRE primary task.

HU PLLR System				Dim	$\%C_{\text{avg}}$	C_{LLR}
Baseline				59+Δ	**2.86**	**0.389**
Supervised	Merge Phones	Family	R	33+Δ	3.07	0.422
			SL	31+Δ	3.46	0.467
			MP	23+Δ	**2.98**	**0.426**
			M	14+Δ	4.22	0.580

(with around the same size as the *Family-R* feature set) suffers higher degradation (3.46 C_{avg}). Instead, the clustering performed according to the *Family-MP* criteria, which provides a feature vector whose size is almost a third of the original baseline vector size, suffers around the same (or less) degradation than the *Family-R* set, attaining 2.98 C_{avg}. Finally, The *Family-M* feature set, reducing the PLLR vector to only 14 dimensions, performs significantly worse than the rest of the approaches, meaning that the number of clusters selected is probably too low, causing a higher loss of information.

Given the results obtained and according to the relation between feature size and performance, *Family-MP* was selected as the best approach among supervised dimensionality reduction (phone merging) techniques.

4.1.2 Unsupervised Techniques

Supervised techniques provide some benefits: they are based on phonetic information or expert knowledge criteria and can be therefore useful approximations *a priori*. At the same time, they pose some problems: knowledge of each language is needed, and therefore studies must be done for each phone decoder; furthermore, there is no freedom to select the dimensionality of the resulting set of phones, given that a predefined set of rules must be applied. Unsupervised techniques instead are more flexible and easily tunable, so that the output dimensionality of the sets of phones can be arbitrarily defined [157], [88]. As a con, the clusters obtained with these techniques can not be easily interpreted. In this work, several clustering approaches were studied considering mutual information, correlation or covariance between phonemes, and several phoneme selection criteria, such as the overall frequency of phonemes, standard deviations of the distributions of the phonemes between languages, etc. Finally the following criteria have been applied:

- *Correlation*: An iterative clustering algorithm is used. In each step, the algorithm merges the closest phone pair (or phone group pair) according to the correlation among the phone posterior probabilities. The clustering provided a singular feature set, with a few sets composed of a relatively high number of phonemes, and a lot of non-grouped phonemes.
- *Frequency*: The N phones with the highest posterior probabilities overall in the training set are selected as most relevant, and therefore used as (reduced) phone set. With this criteria, half of the vowel phonemes were discarded, as well as most of the *long*, affricate and palatal consonants.

Finally, Principal Component Analysis (PCA) [14] was also tested. Since PCA is an orthogonal transformation that is assumed to deal with normally distributed data ranging in $(-\infty, \infty)$, it was not a suitable transformation to be applied on the phone posterior probability space, which ranges in [0,1]. Instead, PCA can be directly applied on the normally distributed PLLR space, which ranges in $[-\infty, \infty]$.

4.1.2.1 Results and Selection of the Optimal Unsupervised Technique

For studying the unsupervised dimensionality reduction techniques, the baseline and the best supervised clustering technique were taken as reference. In Table 4.5, the performance figures attained by those reference approaches and the systems trained on the feature sets obtained with the unsupervised techniques are shown. To provide a fair comparison among approaches, and given that the dimensionality could be easily set in these experiments, unsupervised techniques were configured to provide feature sets of 23 dimensions (matching the size of the *Family-MP* feature set).

As shown in Table 4.5, the system trained on the *Correlation* reduced set of features performs much worse than the one based on the *Family-MP* set (3.76 vs 2.98 C_{avg}). The same happens with the approach trained on the *Frequency* reduced set of features (3.56 C_{avg}). Surprisingly, the feature projection method (PCA) not only outperforms other dimensionality reduction approaches, but it also outperforms the baseline system reaching 2.45 C_{avg}.

4.1.3 Combination of Systems using Different Decoders

This section shows results for the baseline system and the two best dimensionality reduction approaches: supervised *Family-MP* clustering and PCA projection, on different datasets and for different phone decoders, to check the consistency of the conclusions attained. In the set of experiments presented in this section, 10 iterations were applied to compute the Total Variability Matrix.

TABLE 4.5: $\%C_{\text{avg}}$ and C_{LLR} performance for the PLLR i-vector system with different unsupervised dimensionality reduction approaches, on the NIST 2007 LRE primary task.

HU PLLR System				Dim	$\%C_{\text{avg}}$	C_{LLR}
Baseline				59+Δ	**2.86**	**0.389**
Supervised	Merge Phones	Family	MP	23+Δ	**2.98**	**0.426**
Unsupervised	Merge Phones	Correlation		23+Δ	3.76	0.523
	Select Phones	Frequency		23+Δ	3.56	0.480
	PLLR Projection	PCA		23+Δ	**2.45**	**0.333**

4.1.3.1 Results on the NIST 2007 LRE dataset

Results of the baseline and best dimensionality reduction approaches using multiple decoders on the NIST 2007 LRE are shown in Table 4.6.

TABLE 4.6: $\%C_{\text{avg}}$ and C_{LLR} performance for PLLR i-vector baseline system, and systems using PLLR features reduced to the Family-MP set and projected with PCA, for each of the BUT decoders, and their fusion, on the NIST 2007 LRE primary task.

PLLR System		$\%C_{\text{avg}}$	C_{LLR}
Baseline	CZ (43+Δ)	4.18	0.550
	HU (59+Δ)	2.66	0.382
	RU (50+Δ)	4.08	0.549
	CZ+HU+RU	**2.09**	**0.299**
Family-MP	CZ (25+Δ)	4.55	0.619
	HU (23+Δ)	3.08	0.424
	RU (21+Δ)	4.30	0.598
	CZ+HU+RU	**2.24**	**0.313**
PCA	CZ (25+Δ)	3.12	0.432
	HU (23+Δ)	2.17	0.320
	RU (21+Δ)	3.29	0.451
	CZ+HU+RU	**1.79**	**0.240**

Even though the best performance is attained with the HU phone decoder, conclusions are the same with all decoders. The *Family-MP* approach degrades system performance between 6% (RU) and 16% (HU) in terms of C_{avg}. On the other hand, PCA always provides a significant gain with regard to the baseline approach, ranging from 18% (HU) to 25% (CZ) relative improvements in terms of C_{avg}. When fusing the systems using different decoders trained on the *same* set of features (meaning that we use the same baseline or clustering approach), performance improves in all cases. Fusion provides a relative improvement of 21% with regard to the best individual system when fusing the baseline approaches, 27% for the *Family-MP* approaches and 18% when fusing systems trained on the features projected by PCA.

4.1.3.2 Results on the NIST 2011 LRE dataset

Results on the NIST 2011 LRE for the systems trained on the baseline, *Family-MP* and PCA-projected set of features are outlined in Table 4.7.

TABLE 4.7: $\%C_{\text{avg}}$, C_{LLR} and $C_{avg}^{24} \times 100$ performance for the PLLR i-vector baseline system, and systems using PLLR features reduced to the Family-MP set and projected with PCA, for each of the BUT decoders, and their fusion, on the NIST 2011 LRE primary task.

PLLR System		$\%C_{\text{avg}}$	C_{LLR}	$\%C_{avg}^{24}$
Baseline	CZ (43+Δ)	5.31	0.978	12.46
	HU (59+Δ)	5.18	0.982	12.12
	RU (50+Δ)	4.70	0.898	11.27
	CZ+HU+RU	**3.79**	**0.720**	**9.10**
Family-MP	CZ (25+Δ)	5.53	1.054	13.62
	HU (23+Δ)	5.40	1.015	12.64
	RU (21+Δ)	5.13	0.961	11.57
	CZ+HU+RU	**3.82**	**0.693**	**9.79**
PCA	CZ (25+Δ)	4.46	0.855	11.20
	HU (23+Δ)	4.48	0.877	10.88
	RU (21+Δ)	4.20	0.803	11.01
	CZ+HU+RU	**3.21**	**0.634**	**8.45**

Results are consistent with the ones attained on the NIST 2007 LRE. On this dataset the relative degradation of the systems trained on the *Family-MP* set with regard to the ones trained on the baseline features range from 4% (HU) to 9% (RU) in terms of C_{avg}. Still, when fusing the systems trained on different decoders, the performance attained by both sets of fused systems is comparable (3.79 C_{avg} for

the baseline vs 3.82 of the *Family-MP*). Regarding the PCA approach, performance relative improvements range from 11% (RU) to 16% (CZ) in terms of C_{avg} when applying this technique. The fusion of the different PCA systems using the three phone decoders reaches 3.21 C_{avg}.

4.2 PCA Dimensionality Optimization

Given the conclusions attained on the PLLR dimensionality reduction study, we decided to follow the study of the application of PCA on top of the features, to try to get an insight of the behavior of the features when applying this technique. Moreover, we wanted to find how far the dimensionality could be reduced, without getting a high performance loss, in order to (possibly) be able to compute Shifted Deltas on top of PLLRs. Therefore, the two main objectives were:

- To find the best dimensionality to which the PLLR feature set could be projected to optimize performance.
- To find the smallest dimensionality to which the PLLR feature set could be reduced without a high performance degradation, in order to compute shifted deltas on top of the features.

Figure 4.3 shows C_{LLR} and C_{avg} performance for the baseline PLLR system and various systems trained on PLLR features projected by means of PCA to different dimensionalities. Baseline reaches 2.86 C_{avg}. Note that performance is significantly enhanced by simply projecting the features (without dimensionality reduction). This could be explained by the whitening of the data attained when applying PCA, which makes the features more suitable for the diagonal covariance models used in our approaches.

There is a wide range of dimensionalities in which performance is not significantly degraded. Different ranges can be found when analyzing results. The optimal range, that is, the one in which performance is mostly enhanced, is between 47 and 27, best results being attained when using PCA to project PLLR features into 33 dimensions, with 2.11 C_{avg}. Performance degrades slightly after that range, and starts being severe around dimension 15, where a significant loss of information is revealed.

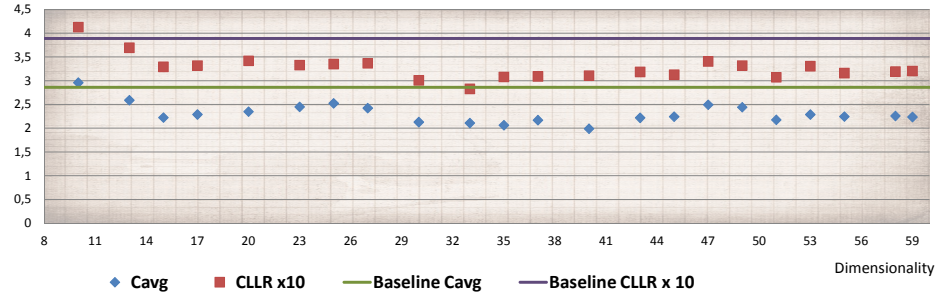


FIGURE 4.3: C_{avg} and $10\times C_{\text{LLR}}$ performance for the PLLR-based baseline system (with feature dimensionality=59) and systems trained on the set of PLLR features obtained after PCA projection into different dimensionalities, on the NIST 2007 LRE primary task.

4.3 Shifted Delta PLLRs

The dimensionality reduction study showed that PLLRs could be reduced to around 15 dimensions without strongly harming performance, which is a manageable dimension to compute Shifted Deltas.

4.3.1 Shifted Delta Parameter Optimization

The study presented in this section focused on checking whether the application of Shifted Delta (SD) on top of PLLRs could enhance system performance. A first series of experiments searched for the parameter N, which sets the number of coefficients from which derivatives are computed at each frame (see 2.2.3 for details on SD configuration parameters). Given that 15 was the dimensionality to which features could be projected before strongly harming performance, values around this number were tested (13, 15 and 17). To begin with, standard configuration parameters were selected for the rest of SD parameters (the ones commonly used in our MFCC-based systems, that is d=2, P=3 and k=7). Results are outlined in Table 4.8.

This first set of figures showing performance of SD-PLLR feature based systems reveal that performance can actually be enhanced by applying SD on top of PLLRs, as the approaches trained with SD-PLLRs outperform the ones trained on PCA-projected PLLR features. Furthermore, they also outperform the best result attained by projecting the features by means of PCA (PLLR-PCA to 33 dimensions, 2.11 C_{avg}). Even though the performance of the PCA-projected PLLR system degraded more noticeably when projecting the features to 13 dimensions (2.59 C_{avg}), this was

TABLE 4.8: SLR performance of an i-vector system based on SD-PLLR features using different N values on the NIST 2007 LRE 30s test set.

PCA Dim		C_{avg}	C_{LLR}
13	PLLR+Δ	2.59	0.370
	SD- PLLR 13-2-3-7	1.71	0.260
15	PLLR+Δ	2.23	0.330
	SD- PLLR 15-2-3-7	1.94	0.264
17	PLLR+Δ	2.29	0.332
	SD- PLLR 17-2-3-7	1.73	0.241

the optimal value found to optimize SD-PLLR system performance with regard to N, reaching 1.71 C_{avg}.

Once confirmed that SD could be applied to extract relevant dynamic information from PLLR features, optimal values for the remaining SD parameters were explored.

TABLE 4.9: SLR performance of an i-vector system based on SD-PLLR features, using different P values, on the NIST 2007 LRE 30s test set.

SD-PLLR configuration	C_{avg}	C_{LLR}
13-2-1-7	2.39	0.346
13-2-2-7	1.91	0.279
13-2-3-7	1.71	0.260
13-2-4-7	2.02	0.297
13-2-5-7	2.46	0.347

Results for systems trained using different values for the shift, parameter P, are shown in Table 4.9. Figures show a high sensitivity with regard to this parameter, optimal performance being found (as usual) for $P = 3$ (13-2-3-7).

TABLE 4.10: SLR performance of an i-vector system based on SD-PLLR features, using different d values, on the NIST 2007 LRE 30s test set.

SD-PLLR configuration	C_{avg}	C_{LLR}
13-1-3-7	2.04	0.286
13-2-3-7	1.71	0.260
13-3-3-7	2.03	0.277

Finally, the parameter d (size of the windows) was also optimized. Values around the standard configuration were tested once again. Performance of systems trained with different d parameter values are shown in Table 4.10. The best results were attained for $d = 2$. Therefore, the 13-2-3-7 configuration was selected as optimal on the NIST 2007 LRE dataset.

4.3.2 Results on NIST 2011 LRE dataset

The optimal configuration found for SD parameters on the NIST 2007 LRE dataset was used on the NIST 2011 LRE. Performance is compared to the baseline trained on standard PLLR features (not projected). Results are shown in Table 4.11. Figures attained reveal that the application of SD on PLLRs provide a a 21% relative improvement in terms of C_{avg} with regard to the baseline.

TABLE 4.11: SLR performance of i-vector systems based on PLLR and SD-PLLR features, on the NIST 2011 LRE 30s test set.

System	C_{avg}	C_{LLR}	$\%C_{avg}^{24}$
Baseline	5.18	0.982	12.12
SD-PLLR 13-2-3-7	4.10	0.826	10.48

4.4 Chapter Summary

We have outlined how the PLLR feature set can be reduced to almost a third of its original size with a little performance degradation, based on a supervised clustering technique applied on the phone posterior space, which still keeps the meaning of the clusters, making it a suitable approach for other possible techniques. Furthermore, the studies presented have revealed that the sole application of PCA can enhance the performance of the systems, probably due to the whitening of the data, making them more suitable to the diagonal covariance models used in our approaches.

The search for an optimal (reduced) dimension has shown that PLLR features perform similarly when projected to a wide range of dimensions, reaching the optimal performance for around 33 dimensions.

Finally, it has been found that shifted deltas are a nice way of introducing larger temporal context information into the feature vector, providing significant performance improvements in both NIST 2007 and 2011 LRE datasets.

Chapter 5

PLLR Feature Projection

The analysis presented in Chapter 4 dealing with several dimensionality reduction issues, posed a couple of questions. First, why was performance so enhanced when applying PCA? It was straightforward to understand that the decorrelation of the features would be providing a significant improvement given the diagonal covariance GMM models that we use in our experiments, yet, some additional underlying reasons could be contributing to the enhancements. Second, why was it possible to reduce PLLR dimensionality so much without degrading performance?

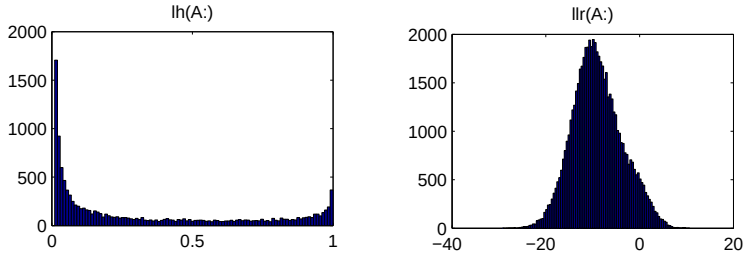
These issues brought us to start a study in order to get a better understanding of the way the PLLRs carry information and their behavior, exploring the multidimensional distribution of the features.

5.1 Analysis of the PLLR Feature Space

To get an insight into the PLLR feature behavior, let's go back to the definition of the features. The way PLLRs are computed from the phone posteriors was defined in Equation 3.4. That is, a PLLR is defined as the logarithm of the posterior probability ratio of a phoneme, normalized by a $\frac{1}{(N-1)}$ term. The normalization term of the definition is useful to get an intuitive idea of the values that PLLR features may take. With the normalization term, a $PLLR > 0$ would be representing a phoneme with a posterior probability higher than the average, under the assumption that the probability mass was equally distributed among all the phonemes. On the other hand, a $PLLR < 0$ would represent a phoneme with a lower posterior probability than the average under the same conditions. In any case, the normalization term

just introduces a constant offset in the value of the features (and has therefore no effect on the results). So, for the sake of clarity, in the remaining studies, let us redefine the features, suppressing the *unnecessary* normalization term $\frac{1}{(N-1)}$. Therefore, PLLRs can be simply redefined as phone posterior logits. Given a phone decoder that outputs an N-dimensional vector of phone posteriors at each frame: $\mathbf{p} = (p_1, p_2, \ldots, p_N)$, such that $\sum_{i=1}^{N} p_i = 1$ and $p_i \in [0, 1]$ for $i = 1, 2, \ldots, N$, PLLRs are now re-defined as:

$$r_i = \text{logit}(p_i) = \log \frac{p_i}{(1 - p_i)} \qquad i = 1, ..., N \tag{5.1}$$



(a) (b)

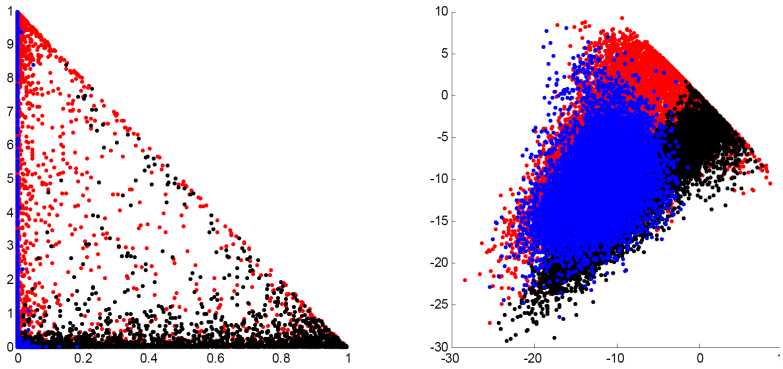
FIGURE 5.1: Distributions of (a) frame-level phone posteriors and (b) frame level phone log-likelihood ratios for the Hungarian phone *a:*.

As exposed in Section 3.1, PLLRs seem to overcome the non-Gaussian nature of phone posteriors for each individual phone model. Figure 5.1 shows an example of the distribution of phone posteriors and PLLR features for the Hungarian phone *a:* (as defined in SAMPA[1]), for a subset of NIST 2007 LRE data. Clearly, phone posteriors show a non-Gaussian distribution, whereas PLLR features are apparently Gaussian distributed.

However, when the distribution of two (or more) PLLRs is analyzed, these seemingly Gaussian distributed features show a strongly bounded shape. Figure 5.2 shows two dimensional distributions of (a) phone posteriors, and (b) PLLRs, for three pairs of phones. These distributions are clearly bounded by a line. Figure 5.3 illustrates the three-dimensional distribution of (a) phone posteriors and (b) PLLRs, for the set of phones (*a:*, *E*, *O*). In this case, PLLRs appear to be bounded by a convex function.

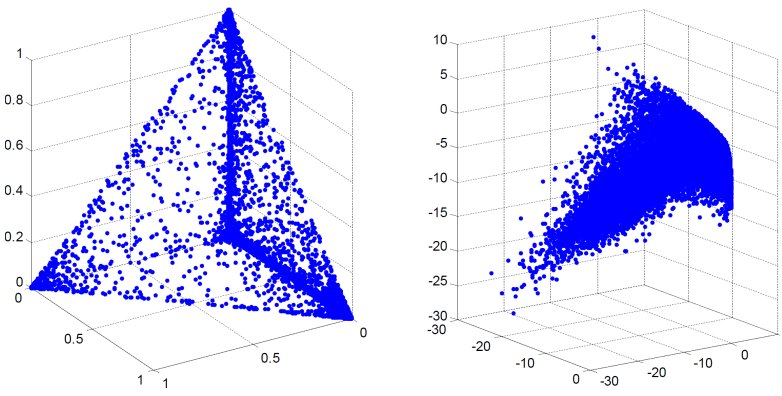
Phone posteriors range in $[0, 1]$, but they must satisfy the second axiom of probability, $\sum_{i=1}^{N} p_i = 1$. As a result, (and as exposed in Section 3.1) phone posteriors

[1] http://www.phon.ucl.ac.uk/home/sampa/hungaria.htm



(a) (b)

FIGURE 5.2: Distributions of (a) phone posteriors and (b) PLLRs, for three pairs of phones, *a:* vs *E* (red), *i* vs *i:* (black) and *dz* vs *h* (blue).



(a) (b)

FIGURE 5.3: Distributions of (a) phone posteriors and (b) PLLRs, for the set of phones (*a:*, *E*, *O*).

lie in an $(N-1)$ dimensional region defined as standard (N-1) simplex, which is the subset of points defined by:

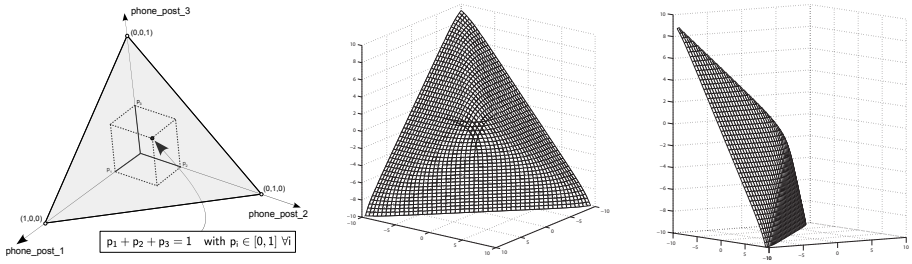
$$\Delta^{(N-1)} = \{\mathbf{p} \in \mathbb{R}^N \mid \mathcal{F}(\mathbf{p}) = \sum_{i=1}^{N} p_i - 1 = 0 \ \wedge \ p_i \geq 0 \ \forall i\} \tag{5.2}$$

where $\mathcal{F}(\mathbf{p})$ is the implicit hyper-plane function.

Given that phone posteriors range in [0,1], PLLRs would seemingly range in $[-\infty, \infty]$, but the constraint among phone posteriors is transferred into the PLLR space. From Equations 1 and 2, we derive the hyper-surface S where PLLRs lie as:

$$\mathcal{S}^{(N-1)} = \left\{ \mathbf{r} \in \mathbb{R}^N \;\middle|\; \mathcal{G}(\mathbf{r}) = \sum_{i=1}^{N} \frac{1}{1+e^{-r_i}} - 1 = 0 \right\} \tag{5.3}$$

where $\mathcal{G}(\mathbf{r})$ is the implicit hyper-surface function.



(a) (b) (c)

FIGURE 5.4: (a) Standard 2-simplex defined by phone posteriors in the case of a phone decoder with 3 phonetic units. Graphs (b) and (c) show the hyper-surface where PLLRs lie for the case of a phone decoder with 3 phonetic units.

Figures 5.4(b) and 5.4(c) show two partial views of the hyper-surface $\mathcal{S}$ for the case of a phone decoder with 3 phonetic units.

The hyper-surface $\mathcal{S}$ is asymptotically perpendicular to the basis of PLLRs, which explains the bounded distributions shown in Figures 5.2(b) and 5.3(b). Let's prove it.

The normal vector to the hyper-surface $\mathcal{S}$ is:

$$\mathbf{n} = \nabla\mathcal{G}(\mathbf{r}) \tag{5.4}$$

where each component n_i of $\mathbf{n}$ is given by:

$$n_i = \frac{e^{r_i}}{(1+e^{r_i})^2} \tag{5.5}$$

Let us consider the case in which a subset of phones $\mathcal{I} \subset \{1, 2, ..., N\}$ accounts for most of the probability mass, that is, $\sum_{\forall i \in \mathcal{I}} p_i = 1 - \epsilon$. As these phones tend to take all the probability mass, it follows that $\sum_{\forall i \in \mathcal{I}} p_i \to 1$ and $\epsilon = \sum_{\forall i \notin \mathcal{I}} p_i \to 0$ (therefore, $r_i \to -\infty \; \forall i \notin \mathcal{I}$). Accordingly, for the normal vector $\mathbf{n}$ it holds:

$$\lim_{\epsilon \to 0} n_i = 0 \qquad \forall i \notin \mathcal{I} \tag{5.6}$$

That is, the normal vector tends to lie in the subspace $\mathcal{Q}$ where the set of phones $\mathcal{I}$ are confined. Hence, the surface is asymptotically perpendicular to any basis defined on $\mathcal{Q}$. Figures 5.4(b) and 5.4(c) illustrate the surface, and its asymptotic behavior, for the case of a phone decoder with three phonetic units.

5.2 Projection of the Features

To avoid the bounding effect described in Section 5.1, we propose to project PLLRs into the hyper-plane tangential to the surface at the point where all the posteriors take the same value $p_i = \frac{1}{N}$, that is, the top of the convex surface[2], where the normal vector is:

$$\mathbf{n}|_{r_i = -\log(N-1)} = \frac{(N-1)}{N\sqrt{N}} \cdot \hat{\mathbf{1}} \tag{5.7}$$

where $\hat{\mathbf{1}} = \frac{1}{\sqrt{N}}[1_1, 1_2, ..., 1_N]$.

By inspecting Figures 5.4(b) and 5.4(c), it can be seen that, for the case of a decoder consisting of 3 phonetic units, the top of the convex surface is normal to $[1, 1, 1]$. The tangential plane at that point is not orthogonal to any of the three asymptotic planes defined by the PLLR surface, meaning that PLLR projections on such plane will not be bounded.

In the general case (N dimensions), the kernel (null space) of the desired projection is $\hat{\mathbf{1}}$, then the matrix P used to project the data into the selected hyper-plane is given by:

$$P = \mathbb{I} - \hat{\mathbf{1}}' * \hat{\mathbf{1}} \tag{5.8}$$

[2]In the case of the original PLLR definition, this point would be located at the origin [0,0,...,0] in the PLLR space. In the case of the new the *logit* re-definition of PLLRs, this point is located at $[\frac{1}{N-1}, \frac{1}{N-1}, ..., \frac{1}{N-1}]$.

Figure 5.5 shows the transformation (projection) of the PLLRs displayed in Figures 5.2(b) and 5.3(b) into the new basis. No bounds can be observed in the distribution of the projected features.

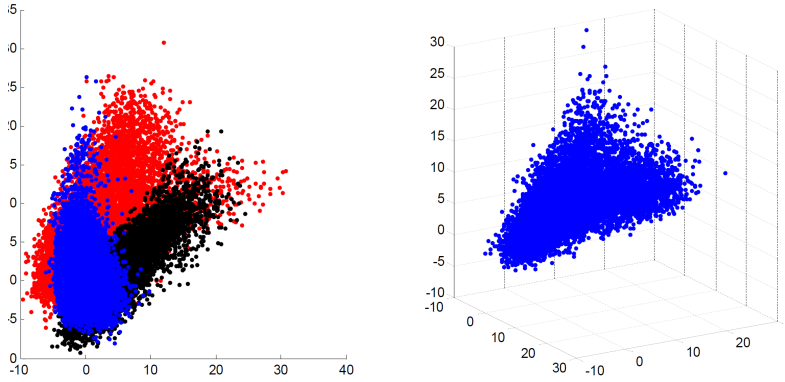


FIGURE 5.5: Distribution of the PLLRs shown in Figures 2(b) and 3(b) after projecting them into the defined hyper-plane tangential to the surface at the point [1/N, 1/N, ..., 1/N].

5.2.1 Feature Decorrelation

Finally, in order to decorrelate the parameters, we apply PCA on the transformed PLLRs, so that they are more suitable for the diagonal covariance Gaussian Mixture Model (GMM) that we are using as Universal Background Model (UBM) in our approaches. Since the projected features lie on an $(N-1)$-dimensional hyper-plane, the number of non-zero eigenvalues of the PCA projection matrix will be $N-1$. Therefore, the dimensionality of the feature vectors, after PCA, will be reduced by one.

5.2.2 Results on the NIST 2007 LRE dataset

Table 5.1 shows results for the baseline system, trained on the original set of PLLR features, and the ones attained for the system trained on the proposed projected features, as well as the figures attained after application of PCA.

The projection of the features clearly enriches the information retrieved by the set of features, as it leads to a 19% relative improvement in terms of C_{avg} with regard to the baseline, reaching 2.31% C_{avg}. The decorrelation of the features helps enhancing

TABLE 5.1: $\%C_{\text{avg}}$ and C_{LLR} performance (and relative improvements) for the PLLR i-vector baseline system, and systems using projected PLLR features on the NIST 2007 LRE primary evaluation task.

PLLR System	Dim.	$\%C_{\text{avg}}$ (r.i.)	C_{LLR} (r.i.)
Baseline	59	2.86	0.389
Projection	59	2.31 (19%)	0.320 (18%)
Projection+PCA	58	**2.10 (27%)**	**0.310 (20%)**
PCA	59	2.24 (22%)	0.321 (17%)
	58	2.26 (21%)	0.319 (18%)

system performance, providing a further 7% relative improvement with regard to the baseline, attaining 2.10% C_{avg}.

More experimentation was performed to check the benefits of the method. It could be the case that PCA would not only decorrelate the feature space, but that as a side effect of the rotation it produces, it may be also suppressing the bounding effects in an unsupervised way. To check this, two experiments were carried out by applying PCA on the original (non-projected) features.

First, PCA was applied on the original (non-projected) features without dimensionality reduction, which does provide a gain comparable to that attained with the projection method, presumably attributed to a combination of both effects: decorrelation and rotation. Applying PCA and reducing the dimensionality to 58 dimensions provides no gain with regard to PCA without dimensionality reduction, that is, reducing one dimension by PCA has not the effect achieved by the projection method. This means that reducing one dimension by PCA is not equivalent to the projection method (because the latter is a singular linear transformation). This can be further confirmed by the fact that the minimum variability direction estimated on the original feature space (the one removed by PCA) is not related to the kernel of the projection method proposed in our approach (i.e. vector $\hat{\mathbf{l}}$ in Eq. 5.7). Actually, the direction removed by the proposed method is closely related to the maximum variability direction identified by means of PCA.

Different results are obtained by applying PCA on the original and on the projected features. The combination of the proposed projection, which ensures the removal of the bounding effect, and subsequent decorrelation by means of PCA yields the best result overall.

5.2.3 Results on the NIST 2009 LRE dataset

The merits of the projected PLLR features were also tested in other NIST LRE benchmarks. Table 5.2 presents results for the baseline system, trained on the original set of features, and the results for the system trained on the projected+PCA PLLR features, as well as figures for acoustic and phonotactic systems, and different fusions of them, to check the complementarity of the new set of features and the possible benefits it could provide compared to the baseline.

TABLE 5.2: $\%C_{\text{avg}}$ and C_{LLR} performance for the PLLR i-vector baseline systems, systems using PLLR projected features, acoustic MFCC and phonotactic systems and fusions of them on the NIST 2009 LRE primary evaluation tasks.

Dataset	System		$\%C_{\text{avg}}$	C_{LLR}
2009 LRE	PLLR	(a)	2.42	0.505
	PLLR+Projection+PCA	(b)	2.19	0.443
	Acoustic MFCC	(c)	2.70	0.535
	Phonotactic	(d)	2.49	0.502
	(c)+(d)		1.67	0.346
	(a)+(c)+(d)		1.48	0.321
	(b)+(c)+(d)		1.42	0.307

As for the NIST 2007 LRE, the system trained on the projected PLLRs performs significantly better than the baseline. The fusion of the acoustic and phonotactic systems with the projected-PLLR system attains a slight improvement with regard to the combination of the acoustic, phonotactic and baseline PLLR systems, more pronounced on when comparing them in terms of C_{LLR}.

5.2.4 Results on the NIST 2011 LRE dataset

A comparison of SLR performance of systems trained on different sets of PLLR features for the NIST 2011 LRE is shown in Table 5.3. Fusions of PLLR systems with baseline acoustic and phonotactic systems are shown too.

The relative improvement attained with the projection of the features is pronounced in this benchmark, obtaining a 17% relative improvement with regard to the baseline. Regarding fusions, once again, the fusion of the acoustic, phonotactic and projected PLLR system gets a remarkable improvement with regard to the fusion of the acoustic and phonotactic systems with the baseline PLLR system, a relative 9% in terms of C_{avg}.

TABLE 5.3: $\%C_{\text{avg}}$ and C_{LLR} performance for the PLLR i-vector baseline systems, systems using PLLR projected features, acoustic MFCC and phonotactic systems and fusions of them on the NIST 2011 LRE primary evaluation tasks.

Dataset	System		$\%C_{\text{avg}}$	C_{LLR}	$\%C_{\text{avg}}^{24}$
2011 LRE	PLLR	(a)	5.18	0.981	12.12
	PLLR+Projection+PCA	(b)	4.30	0.824	11.33
	Acoustic MFCC	(c)	5.95	1.088	13.56
	Phonotactic	(d)	7.15	1.280	14.28
	(c)+(d)		4.34	0.823	10.43
	(a)+(c)+(d)		3.63	0.714	9.14
	(b)+(c)+(d)		3.33	0.667	8.91

5.3 Projected PLLRs in Noisy Environments

In collaboration with the Brno University of Technology, the projected PLLR features were tested on the challenging noisy RATS benchmark [117].

Given that despite the RATS evaluation program comprised 120, 30, 10 and 3s segments, LDC only provided 120s signals for training and testing (see Section 2.1 for details), this study [116] made use of specific training and development sets constructed by making cuts of 120s signals, where *main training*, *extended training*, *development* and *calibration* sets were defined [99].

Two hybrid NN/HMM phone decoders were used to estimate three state frame-by-frame posteriors. The two decoders were trained for Levantine Arabic (LE) and Czech, using RATS LE keyword search data and Czech CTS data, respectively, including data corrupted with noise at 10dB level, respectively. The phone decoders feature 36 (LE) and 38 (CZ) phonetic units.

To compute the PLLRs, first the states corresponding to the same phonetic unit were merged, following 3.1. PLLRs were computed according to Equation 5.1, augmented with first order deltas, and projected using the procedure presented in Section 5.2, leading to a 74 dimensional feature vector for CZ and a 70 dimensional feature vector for LE.

The systems were based on an i-vector approach, using a diagonal covariance 2048 dimensional GMM-UBM, and the total variability matrix was trained on the main training set, featuring 600 dimensional i-vectors.

Two different classifiers were used then to build two different systems: Logistic Regression (LR) and Neural Networks (NN). The multiclass regularized logistic regression classifier (based on [14, 18]) was trained on the main training set with within-class covariance normalization conditioned i-vectors. The NN classifier was based on a three layer NN, with 300 neurons in the hidden layer and 6 outputs (5 target + 1 non-target languages). The NN was trained on the extended training set.

Logistic regression calibration parameters were estimated on the calibration set.

5.3.1 Baseline Systems

The acoustic baseline system was based on the i-vector approach, using 20 cepstral PLP2 coefficients [63, 90] augmented with Δ and $\Delta\Delta$s, obtaining a 60 dimensional feature vector. The system used the same configuration and training sets as the ones used by the PLLR-based system, that is, diagonal covariance 2048 component GMM-UBM and 600 dimensional i-vectors.

The phonotactic system was based on the Subspace n-gram Modeling (SnGM) approach with a 600 dimensional subspace estimated over trigram counts trained using regularized multinomial space, as described in [142]. Hard pruning of low-frequency trigrams was applied to reduce problems caused by data sparsity. 600 dimensional i-vectors were estimated, as point estimates of latent variables representing the input utterance dependent n-gram model.

Both systems were used also to train LR and NN classifiers. Logistic regression calibration parameters were estimated on the calibration set.

5.3.2 Results on the RATS dataset

Results attained with all the systems under the LR-classifier approach are presented in Table 5.4. Figures show that both LE and CZ PLLR systems outperform the PLP2-based i-vector approach. Also, PLLR systems outperform their respective phonotactic approaches (that is, the one based on the same phone decoder). Overall, the PLLR-LE system attains the best performance in most conditions, that is, 120s, 30s and 10s, and is only beaten (by a small margin) on the 3s condition by the PLLR-CZ system.

Performance of the systems based on NN classifiers are shown in Table 5.5. The approaches based on NN outperform in all cases the systems based on LR classifiers. Performance figures are consistent with those attained using the LR classifier. Once again, PLLR systems outperform the PLP2-based i-vector approach, and also the

TABLE 5.4: $\%C_{\text{avg}}$ performance for the PLP2, SnGM and PLLR systems with logistic regression classifiers on the RATS evaluation set for the 120s, 30s, 10s and 3s signals.

System	120s	30s	10s	3s
PLP2	7.72	11.69	16.39	23.04
SnGM-LE	5.86	12.28	18.53	26.45
SnGM-CZ	8.59	15.76	20.89	27.95
PLLR-LE	**4.56**	**7.98**	**12.61**	21.48
PLLR-CZ	6.95	10.76	15.13	**21.32**

phonotactic approaches in most cases. PLLR-LE is the system achieving the best performance in all conditions.

Regarding fusions, the fusion of the two PLLR systems attains good results in all conditions, as well as the fusion of both LE phone decoder-based systems. The fusion of the LE systems with the PLP2 approach attains a significant gain for short duration utterances (10s and 3s). Finally, the fusion of all systems, still provides some slight improvements in some conditions.

TABLE 5.5: $\%C_{\text{avg}}$ performance for the PLP2, SnGM and PLLR systems with neural network classifiers on the RATS evaluation set for the 120s, 30s, 10s and 3s signals.

System		120s	30s	10s	3s
1	PLP2	7.21	9.21	12.43	18.58
2	SnGM-LE	5.53	9.34	15.61	22.76
3	SnGM-CZ	7.23	10.46	15.38	24.05
4	PLLR-LE	**5.37**	**7.31**	**11.46**	**17.63**
5	PLLR-CZ	5.81	8.83	12.30	19.52

Fusions	120s	30s	10s	3s
4+5	5.19	6.79	10.14	16.04
1+4	5.80	6.43	8.69	15.37
2+4	**5.12**	6.61	10.48	16.93
3+5	5.74	8.33	11.28	18.11
1+2+4	5.38	6.31	**8.53**	14.90
1+4+5	5.29	6.43	8.71	14.65
1+2+3+4+5	5.59	**6.21**	8.75	**14.37**

5.4 Shifted Delta Projected PLLRs

Given the results attained with systems trained on the reduced set of PLLR features augmented with Shifted Deltas, the same transformation was applied on top of the projected set of PLLR features, to check the possible enhancement it could provide.

5.4.1 Results on the NIST 2007 LRE dataset

TABLE 5.6: $C_{\text{avg}} \times 100$ and C_{LLR} performance for the Projected PLLR + PCA, Projected PLLR + PCA reduced and Projected PLLR + PCA reduced + SD approaches on the NIST 2007 LRE primary evaluation tasks.

PLLR System	$\%C_{\text{avg}}$	C_{LLR}
Projection + PCA 58	2.10	0.310
Projection + PCA 13	2.43	0.330
Projection + PCA 13 + SD	1.52	0.225

Table 5.6 shows results of the system trained with the projected features, the one trained with the projected and reduced set of features, and the one trained with projected, reduced and augmented with SD features on the NIST 2007 LRE dataset. As expected, performance degraded when reducing the dimensionality of the set of features, from 2.10 to 2.43 in terms of C_{avg}, and significantly enhanced when using SD, reaching 1.52 C_{avg}.

Figure 5.6 represents graphically the performance of the baseline system in terms of C_{avg} and improvements attained at each step, that is, the gain attained when projecting the features (a relative 19%), after the decorrelation of the parameter space (a further 7% with regard to the baseline) and after augmenting the features with SD (another 20%). The system trained with the fully transformed features (after the three step process) attains an overall 47% relative improvement with regard to the system trained with the original PLLR features.

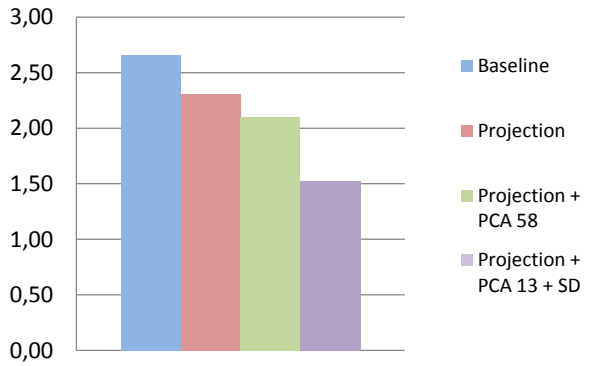


FIGURE 5.6: $C_{\text{avg}} \times 100$ performance for the Baseline (blue), Projected PLLR (red), Projected PLLR + PCA 58 (green) and Projected PLLR + PCA 13 + SD (purple) approaches.

5.4.2 Results on the NIST 2011 LRE dataset

Table 5.7 shows results of the system based on projected+SD features on the NIST 2011 LRE dataset. Comparing the results attained by the SD-PLLR based system, using 13-2-3-7 SD parameter configuration with those attained by the system trained on the projected PLLR features, SD-PLLR features appear to be less informative. All the experimentation performed with the features in previous studies has shown that SD computation on top of PLLRs is beneficial to improve system performance. Therefore, the intuition for the reason behind these results was that the dimensionality reduction applied on top of the projected features was degrading system performance further than in other datasets, and that SD application was not improving performance enough to recover that loss. With the aim of testing this fact, other two systems were trained, based on SD-PLLR features, but computed from PLLRs reduced to not so small dimensions, using 15-2-3-7 and 17-2-3-7 SD parameter configurations. The figures attained by the SD-PLLR based system using 15-2-3-7 SD parameter configuration range close to those of the baseline system. The system trained on SD-PLLR features with 17-2-3-7 SD parameter configuration outperforms the baseline, reaching 3,80% in terms of C_{avg}, which is the best result attained by a single system for this benchmark, and confirms that the PCA dimensionality reduction should be optimized for each dataset, before SD computation.

TABLE 5.7: $C_{\text{avg}} \times 100$, C_{LLR} and $\%C_{\text{avg}}^{24}$ performance for the Projected PLLR + PCA, and Projected PLLR + PCA reduced + SD approaches for 13, 15 and 17 dimensionalities on the NIST 2011 LRE primary evaluation tasks.

PLLR System	$\%C_{\text{avg}}$	C_{LLR}	$\%C_{\text{avg}}^{24}$
Projection + PCA 58	4.30	0.824	11.33
Projection + PCA 13 + SD	4.84	0.916	12.49
Projection + PCA 15 + SD	4.39	0.833	11.26
Projection + PCA 17 + SD	3.80	0.756	9.52

5.5 Chapter Summary

The projection of PLLR features has been found as an effective way of extracting relevant information for SLR tasks, providing significant performance improvements in all benchmarks.

The study of the use of PLLR features (under different approaches) in noisy environments has revealed that these features are robust against channel mismatch and noisy conditions.

Finally, the application of shifted deltas on top of the projected features has also brought further improvements in performance, confirming that the use of dynamic information in large temporal contexts (regardless of the feature representation) provides relevant information for SLR.

The studies performed in previous chapters, have focused on the search for an optimal PLLR extraction procedure. According to the results attained, to summarize, the optimal approach for PLLR computation would comprise:

- Selection of a phone decoder with high phonetic coverage
- Estimation of PLLRs as defined in Equation 5.1
- PLLR feature projection following the procedure described in Section 5.2, using the projection matrix defined in Equation 5.8
- PCA projection performing dimensionality reduction optimization for the database
- SD computation on top of the projected and reduced set of PLLRs

Chapter 6

PLLRs for Speaker Recognition

Speaker and language recognition are closely related pattern recognition tasks that share not only the main structure of the systems, but, as it was outlined in Chapter 2, many processing aspects such as features, modeling techniques, scoring procedures, etc. Despite all the similarities, common methods and mutual resemblances that can be found among them, there are yet significant differences, which make both tasks challenging in different ways.

- The amount of training data that can be gathered for different speakers is not comparable to the training data that can be found for different languages. It is therefore straightforward that the training stages must differ significantly, as the speaker related tasks pose an extra difficulty for not having that much data (meaning *positive samples*) to rely on.
- Feature extraction, even when based on the same kind of features, uses different tunings and configurations on the extraction stage, as the information that is willing to be found is speaker-specific information. That is, features related with the speaker physiological characteristics such as vocal folds, length and shape of the vocal tract, pitch, energy, etc. Other high-level characteristics are related to the personal lexicon, accent, pronunciation, etc. that could help differentiating them from others. These high level features are not very used in the literature given the level of complexity that the extraction implies.
- Language Recognition has been usually treated as a multi-class recognition task while speaker recognition is established in most challenges as a binary

decision task, which makes a difference in the scoring and backend stages as well as in the metrics used to evaluate the systems.

Besides, both tasks are conceptually different, given that speaker recognition systems are usually trained with data of speakers different to those of the test set, whereas language recognition systems normally rely on data of the languages they are going to be tested with. A language recognition task aiming to recognize a language without training data available for it would render the usual SR scenario in a SLR task, as proposed in the Albayzin 2012 LRE [123].

This Chapter explores the possible benefits of using PLLR features in a speaker recognition task. To that purpose, first, we provide a short overview of state-of-the-art speaker recognition, focusing on the differences with regard to spoken language recognition. Then, we proceed to check test the performance of PLLR features in SR tasks.

6.1 State-of-the-art Speaker Recognition

This section will cover state-of-the-art Speaker Recognition (SR) techniques not shared with spoken language recognition. Details are provided for datasets, feature extraction, modeling techniques and evaluation metrics.

6.1.1 Datasets

The available data for speaker recognition has grown considerably in terms of number of datasets and also regarding database size, specially considering the number of speakers per database. LDC has a relevant collection of databases built for this purpose, which date from 1993. Great efforts have been made in terms of data collection, from those first datasets used on the nineties, which involved around 2400 conversations among 500-600 speakers, to the ones that can be found nowadays, amounting to tens of thousands signals and thousands speakers, and involving test sets with upper bounds rounding 10^6 or even 10^8 trials. There are plenty of resources in different languages designed for SR tasks [2, 102].

NIST SRE Benchmarks

As happened for SLR, NIST also contributed significantly to the advances of SR providing benchmarks and challenging evaluations in a regular basis to the research

community. Starting in 1997, NIST organized speaker recognition evaluations yearly until 2006, and every two years ever since, until 2012.

NIST SRE started focusing only on speaker detection tasks, which would involve either same handset or different handset test trials and test segments of 3, 10 or 30s. In the following years the evaluations evolved, not only in terms of the amount of training data and test segments provided, which kept growing over the years, but also in terms of evaluation tracks and *challenges*.

In 1997 speaker detection was introduced, in which data segments would include conversational speech from telephone calls. Speaker tracking was introduced in 1999, in which participating groups had to detect a specific speaker as a function of time. In 2000, speaker segmentation was launched as a new task, in which several (an unknown number of) speakers had to be identified and tracked in each test segment.

Other interesting conditions were also evaluated over the years, which would vary the amount of available training data for each target speaker, speaker detection in speech signals of other languages, etc. In 2002, speaker detection tests started covering multi-device and/or multi-channel conditions. Besides telephone data, interviews (such as broadcasts and meetings) were also used for the speaker segmentation tracks.

In 2003, SRE finally concentrated simply on speaker detection tasks, with distinctions between limited data and extended data training. After that, evaluations focused on the same task, providing several training and test conditions, with one as the *core test* mandatory track for all participants and others as optional. The conditions covered different durations for training and test utterances, different number of single channel conversation sides for each speaker (2004), and two-channel or summed channel conversations (2005, 2006). The NIST 2008 SRE included, besides the usual telephone conversational excerpts recorded over telephone channels, telephone data recorded over microphone channels and conversational speech data from interviews recorded over room microphone channels. The inclusion of this kind of data also allowed a new evaluation track, involving longer utterances recorded from microphone channels. In 2010, NIST introduced high vocal effort and low vocal effort speech.

In the last SRE, carried out in 2012, even though the main task remained the same, several factors changed compared to previous evaluations. For the first time, knowledge of all target trials was allowed to obtain the trial score. Besides, new tracks were included, which made a difference among known and unknown test trials, that is, test segments for which the system could assume that the non-target trials were produced by known or unknown speakers. In this evaluation, some test segments included also additive noise, which posed a new demanding challenge.

MOBIO

MOBIO is a challenging audio and video database created on collaboration between six sites in five different countries [91, 101], containing speech from 152 speakers, covering both spontaneous and non-spontaneous speech captured by mobile devices (mobiles and laptops). A system based on PLLR features was trained and presented to the speaker recognition MOBIO 2013 (for details see Appendix B).

6.1.2 Feature Extraction

Most SR systems are based on short-term spectral low-level features, which model vocal-tract properties and the spectral envelope of the sounds [86]. Around the nineties, when the use of high level features gained strength in SLR, numerous works were carried out with the aim of making use of phonotactic information also for speaker recognition, and loads of efforts were made to optimize these features for SR based on the idea that different speakers can be characterized by their vocabulary, accent, pronunciation and other linguistic and speaker dependent behavioral features [86].

Systems trained on these high level features proved to provide complementary information [119], but didn't reach the performance that systems based on low-level acoustic features can attain in SR tasks. In the literature, there can be found studies about speaker specific vocabulary [58], the application of phone/word n-grams obtained from phonetic/word decoders [25], and more recently the use of prosodic features [11, 87]. Nowadays, MFCC still stand out as the most common representation [23], though competitive state-of-the-art systems tend to rely on the combination of several low-level features [60, 72].

6.1.3 Modeling

Speaker and language recognition tasks share most of the modeling techniques. Actually, most of the spectral feature based modeling techniques applied in language recognition have been historically introduced for SR tasks [82], [80], [40], and then adapted for SLR. This way, the modeling and channel compensation techniques presented in sections 2.3 and 2.4, such as GMM-UBM, SVMs, NAP, eigenchannels, JFA or i-vectors, are extensively used also for speaker recognition.

Despite the similarities, some modeling approaches have been specifically developed and largely applied to speaker recognition. That's the case of Probabilistic Linear Discriminant Analysis (PLDA).

Probabilistic Linear Discriminant Analysis

The Gaussian PLDA approach, introduced for face detection in [118], aims to separate the undesired variability contained in the observed data and model only the desired variability information to perform recognition. In our case, an observation j corresponding to an i-vector of a speaker i is supposed to be modeled by:

$$\mathbf{w}_{ij} = \mathbf{V}\mathbf{y}_i + \mathbf{U}\mathbf{x}_{ij} + \mathbf{z}_{ij} \tag{6.1}$$

where:

$$\mathbf{y}_i \sim \mathcal{N}(0, \mathbf{I}) \tag{6.2}$$
$$\mathbf{x}_{ij} \sim \mathcal{N}(0, \mathbf{I}) \tag{6.3}$$
$$\mathbf{z}_{ij} \sim \mathcal{N}\left(0, \mathbf{D}^{-1}\right) \tag{6.4}$$

where $\mathbf{D}$ is a diagonal precision matrix and the hidden variables $\mathbf{y}_i$ and $\mathbf{x}_{ij}$ are the speaker and channel factors, respectively [15], while $\mathbf{z}_{ij}$ is the noise term accounting for the rest of the variability. The model $\mathcal{M} = (V, U, D)$ is estimated by EM.

Extensive work has been made to optimize PLDA for SR. For instance, heavy tailed distributions were introduced to try to deal with some problems produced by outliers when using Gaussian modeling [81]. In [24], discriminative training for PLDA function parameters is introduced. In [153] the effect of using a common speaker space distribution for all channels, but channel dependent channel space distributions, has been also explored.

6.1.4 Evaluation Metrics

This section will briefly describe the evaluation measures that are typically used in SR. The EER and DET curves which have already been presented as metrics in state-of-the-art SLR (see Section 2.7 for details), are evaluation metrics commonly used in (and actually more suitable for) SR tasks. The new metrics presented in this section are based on measures presented in section 2.7 and terminology will be used accordingly.

Detection Cost Function (ActDCF)

The detection cost function has been the main evaluation measure in speaker detection tasks from NIST evaluations [4]. The metric is defined as a weighted sum of miss and false alarms, as follows:

$$C_{Det} = C_{miss} \times P_{miss} \times P_T + C_{fa} \times P_{fa} \times (1 - P_T) \tag{6.5}$$

To get a meaningful value (easy to understand and compare), C_{Det} is normalized by the best (lowest) cost it could be attained without processing the trials:

$$C_{Default} = min \left\{ \begin{array}{l} C_{miss} \times P_T \\ C_{fa} \times (1 - P_T) \end{array} \right\} \tag{6.6}$$

$$C_{Norm} = C_{Det}/C_{Default} \tag{6.7}$$

NIST evaluations used as cost model parameters C_{miss}=C_{fa}=1 and P_T=0.001 until 2008. In NIST 2010, cost model parameters were set to C_{miss}=10, C_{fa}=1 and P_T=0.01, focusing system performance on a low false alarm error region.

Primary Cost Function in 2012 SRE

In 2012 SRE, NIST proposed a new metric dependent on the a priori probabilities for the known/unknown non-target speakers, that took into account the cost functions attained at both operating points: the old values used until 2008 (α_1) and the new parameter cost values defined in 2010 (α_2) [5], in this way:

$$C_{Det} = C_{miss} \times P_{miss} \times P_T + C_{fa} \times (1 - P_T) \times (P_{fa|K} \times P_K + P_{fa|NK} \times P_{NK}) \tag{6.8}$$

where P_K is the a priori probability for a non-target speaker to be one of the evaluation target speakers (and $P_{NK} = 1 - P_K$). The final metric is defined as:

$$C_{primary} = \frac{C_{Norm}(\alpha_1) + C_{Norm}(\alpha_2)}{2} \tag{6.9}$$

6.2 Phone Posteriors for Speaker Characterization

Several works have explored how to make use of high level features for speaker recognition In the 2003 Johns Hopkins University (JHU) Summer Workshop, an extensive study was made with the aim of exploiting these features for SR [119], with results not as successful as in language recognition (compared to acoustic approaches in SR). Phone decoder phoneme posterior features have been used to build n-gram models also for speaker recognition [25]. Other works aiming to employ high-level information are also focusing on prosodic features for SR [39, 85, 138].

With regard to the use of phonetic features for SR, in a work where the merits of multilayer perceptron based phoneme recognizer features for a SLR system are presented [154], it is stated that phoneme posterior features are speaker independent if enough amount of all kinds of speakers are used in the training stage. However, each speaker, due to unique physical characteristics, produces speech sounds in a different way. Speech disfluencies are also distinctive of each speaker, and can be used for discrimination. A phone decoder can be seen as a reference system for representing the speech sounds of any speaker in terms of the activation of its phonetic units, which include both static and dynamic information that would help catching the subtle differences between sounds uttered by each speaker. The goodness of this representation will depend on the richness of the inventory of phonetic units handled by the phone decoder, which can be selected regardless of the spoken language.

To check whether speaker dependent information is present in phonetic posteriors (or how much), a proof-of-concept experiment was carried out, using the TIMIT dataset [161] and the open software TRAPs/NN phone decoder for Hungarian, developed by the Brno University of Technology (BUT) [137].

Figure 6.1 shows the comparison of phone usage/recognition for a set of 7 speakers uttering the same English sentence. The Figure shows usage differences among speakers in some phones, covering different kinds of sounds including sets of consonants and vowels. To compute the values represented in the Figure, the following procedure was used:

Let $p(i|t,s)$ be the phone posterior probability of a phone model i $(1 \leq i \leq N)$ at the frame t, for a sentence uttered by speaker s. First, phone posteriors of each phone were averaged:

$$p(i|s) = \frac{1}{T(s)} \sum_{t=1}^{T(s)} p(i|t,s) \tag{6.10}$$

where $T(s)$ is the number of frames of the sentence uttered by speaker s. Next, as some phonemes are more frequent than others (e.g. vowels would be more frequent than consonants), phone usages were scaled to the same range, obtaining normalized average posteriors for the set of speakers:

$$\hat{p}(i|s) = \frac{p(i|s)}{\sum_{s=1}^{S} p(i|s)} \tag{6.11}$$

where S is the number of speakers.

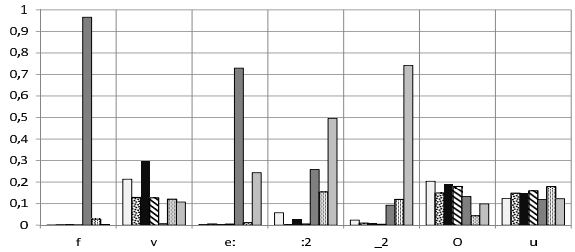


FIGURE 6.1: Normalized average posteriors $\hat{p}(i|s)$ (see Equation 6.11) of seven Hungarian phones on the same utterance (sx9) of the TIMIT dataset for 7 different speakers. The subset of phones represents fricative labiodental consonants (f and v) and a subset of vowels (e:, :2, _2, O, u), as defined in the International Phonetic Alphabet.

Figure 6.1 reveals significant usage differences for some phones. When comparing the usage of consonants, results suggest that one speaker tends to use "f" over "v". Regarding the vowels, there seems to be a set of 2-3 speakers that overuse or prolongate the vowels "e:", ":2" and "_2" with regard to the rest of speakers, whereas the vowels "O" and "u" are almost equally used/pronounced by all of them. These could be intrinsic characteristics of the speakers and could therefore be used for identification.

6.3 Experimental setup

In this section system configuration details are provided.

Datasets

For the SR experiments with PLLR-based systems, two benchmarks were used: the NIST 2010 and 2012 SRE datasets.

Both datasets comprised 5 evaluation tracks (each), containing data recorded/transmitted in different ways. Here is a summary of the core conditions of both SRE:

NIST 2010

1. Interview, same microphone in training and test
2. Interview with different microphone in training and test
3. Interview training, telephone test
4. Interview training, telephone test recorded over microphone
5. Telephone in training and test

NIST 2012

1. Interview with no added noise
2. Telephone with no added noise
3. Interview with added noise
4. Telephone with added noise
5. Telephone recorded in noise

Database configuration details are provided in Appendix A. Given the noisy nature of the signals for conditions 1, 3 and 4 in NIST 2012 SRE, and considering that no noise reduction technique was applied to the signals, for this dataset experimental results are only shown for core conditions 2 and 5, where results are more likely to be reliable (and less degraded than in other conditions).

PLLR Feature Extraction

As for the spoken language recognition experiments, the BUT TRAPs/NN phone decoder for Hungarian was used to compute the PLLR features. Voice activity detection was performed by removing the feature vectors whose highest PLLR value corresponded to the integrated non-phonetic unit (see section 3.2 for details).

MFCC Feature Extraction

MFCC features were computed with a standard configuration for SR systems. Frames of 25 ms at intervals of 10 ms were considered to estimate 13 MFCC coefficients, including the zero (energy) coefficient. Cepstral Mean Subtraction (CMS) and Feature Warping [104] were applied on cepstral coefficients. The feature vector was augmented with dynamic coefficients (first-order and second-order deltas), resulting in a 39-dimensional feature vector. Voice activity detection was performed as in the PLLR system, using the integrated non-phonetic PLLR unit.

i-vector PLDA Configuration

No special treatment was applied to the signals containing additive noise, the Qualcomm-ICSI-OGI (QIO) [10] noise reduction technique (based on Wiener filtering) was independently applied to all audio streams.

Gender dependent 1024-mixture GMMs were trained as UBMs, with diagonal covariance matrix and using binary mixture splitting, orphan mixture discarding and variance flooring. Gender dependent 500 dimensional Total Variability matrices were estimated for each system on each training set. The i-vectors were centered, whitened and length-normalized [64].

A standard PLDA modeling approach was used, where gender dependent PLDA systems [16] were estimated using a speaker subspace of size 150, a channel subspace of size 400 and 20 Expectation Maximization/Minimum Divergence (EM-MD) iterations.

PLDA outputs were directly used as scores for the NIST 2010 experiments.

For the NIST 2012 SRE, scores were post-processed, given that knowledge of other target speakers could be used on the scoring stage. PLDA system scores $s(u,t)$ corresponding to a test utterance u and a training signal t were interpreted as log-likelihoods, that is: $s(u,t) \equiv \log p(u \mid t)$, and the likelihood $p(u \mid i)$ for a speaker i was computed as the average likelihood over all the training signals of that speaker [49]:

$$p(u \mid i) = \frac{1}{|\text{Train}(i)|} \sum_{t \in \text{Train}(i)} e^{s(u,t)} \tag{6.12}$$

Finally, verification scores were computed as closed-set log-likelihood ratios:

$$s(u,i) = \log \frac{p(u \mid i)}{\frac{1}{N-1}\sum_{j \neq i} p(u \mid j)} \tag{6.13}$$

Fusion and Calibration

The BOSARIS toolkit [19] was used to estimate and apply calibration and fusion parameters. The whole training set was used for the estimation of calibration and fusion parameters.

Given that each evaluation had different application costs, calibration parameters were optimized for each dataset. For NIST 2010 SRE experiments, the system was calibrated using $P_{\text{fa}} = 0.01$, $C_{\text{miss}} = 10$, $C_{\text{fa}} = 1$. For NIST 2012 SRE experiments, the system was calibrated using $P_{\text{fa}} = 0.001$, $C_{\text{miss}} = 1$, $C_{\text{fa}} = 1$.

6.4 Search for the Optimal PLLR Feature Configuration

As for spoken language recognition, a development study was carried out to optimize the PLLR feature extraction parameters for SR. The study was performed on the NIST 2010 SRE dataset.

Dynamic coefficients

First of all, the effect of dynamic coefficients was tested on top of the PLLRs. Table 6.1 shows results for the system trained only on PLLR features, PLLRs augmented with first order (Δ) dynamic coefficients and PLLRs augmented with first and second order dynamic coefficients ($\Delta\Delta$).

TABLE 6.1: MinDCF performance of systems using only PLLR features and PLLR features augmented with dynamic coefficients on the NIST 2010 SRE core conditions.

SYSTEM	Core Condition				
	(1)	(2)	(3)	(4)	(5)
PLLR	0.683	0.811	0.889	0.697	0.864
PLLR+Δ	**0.653**	**0.804**	**0.862**	**0.690**	**0.848**
PLLR+Δ+Δ^2	0.702	0.834	0.904	0.745	0.852

As in the case of SLR, the use of first order deltas improves system performance, whereas the use of second order coefficients degrades results. In experiments reported below, PLLR+Δ are therefore used in all PLLR systems.

Variability compensation

Table 6.2 shows the performance of the PLLR-based system using different variability compensation techniques at the feature extraction stage, that is, RASTA filtering, Feature Warping (FW) and Mean and Variance Normalization (MVN). Except for the case of FW in the core condition 5 (telephone in training and test), none of the techniques outperforms the baseline system trained only on PLLR features with no variability compensation technique applied. Therefore, none of the techniques studied in this section will be used when computing the features in the experiments reported below.

TABLE 6.2: MinDCF performance of systems using PLLR features under different configurations on the NIST 2010 SRE core conditions.

SYSTEM	Core Condition				
	(1)	(2)	(3)	(4)	(5)
PLLR+Δ	**0.653**	**0.804**	**0.862**	**0.690**	0.848
(PLLR+Δ)+RASTA	0.830	0.903	0.951	0.892	0.982
(PLLR+Δ)+FW	0.740	0.859	0.883	0.757	**0.794**
(PLLR+Δ)+MVN	0.735	0.852	0.869	0.750	0.821

PLLR feature projection

Table 6.3 presents results of the systems based on PLLR and projected PLLR features.

TABLE 6.3: MinDCF performance of systems using PLLR and projected PLLR features on the NIST 2010 SRE core conditions.

SYSTEM	Core Condition				
	(1)	(2)	(3)	(4)	(5)
PLLR+Δ	**0.653**	**0.804**	**0.862**	**0.690**	**0.848**
Projected PLLR+Δ	0.671	0.823	0.912	0.802	0.913

Unlike for SLR, in SR the projection of the features does not enhance the performance of the system. This fact, must be related to the information contained in the direction [1,1,...,1] that is "suppressed" when projecting the features, the one related to the normal vector of the hyperplane to which the features are projected (see Section 5.2 for details). Given that the projection is a reversible transformation, the information remains in all the other directions after the projection (the info is not discarded nor deleted), but it cannot be directly modeled. The inability to directly model that information seems to be disadvantageous for SR.

6.5 Overall Performance of PLLR Based Systems

This section presents and briefly analyzes results for PLLR based systems using the optimal configuration discussed in the previous section, on the NIST 2010 and 2012 SRE datasets.

6.5.1 Results on the NIST 2010 SRE dataset

First, the PLLR system was tested on the NIST 2010 SRE dataset, compared and fused with the MFCC-based system. Figures attained are shown in Table 6.4. The performance of the PLLR-based i-vector system is far from the one attained by the MFCC-based i-vector system. Depending on the condition, the EER doubles or even triples the one attained by the acoustic system. Nevertheless, quite good performance is attained by the PLLR-based system compared to those usually attained by phonotactic systems on SR tasks [25].

When focusing on the results attained by the fusion of both systems, the contribution of PLLRs is remarkable, as the help improving MinDCF performance in all conditions with a relative improvement with regard to the MFCC-based system ranging from 7% to 16%. They also help improving EER in all but core condition 4, providing up to a 25% relative improvement (core condition 1).

6.5.2 Results on the NIST 2012 SRE dataset

The same tests were carried out on the NIST 2012 SRE dataset to check the contribution of the PLLR-based i-vector system. Results are shown in Table 6.5 for core conditions 2 and 5, which involve telephone recordings with no added noise and recorded in noise, respectively. Results are similar to those attained on the NIST 2010 SRE dataset. The performance of the PLLR-based system is worse than the one of the MFCC-based system, but their fusion provides a gain with regard to the

TABLE 6.4: Results of i-vector /PLDA SR systems based on MFCC and PLLR features, and the fusion of them, on the NIST 2010 SRE core conditions.

Condition		System	EER	MinDCF	ActDCF
(1)	Interview same microphone in training and test	MFCC	1.86	0.417	0.439
		PLLR+Δ	4.05	0.653	0.854
		Fusion	**1.40**	**0.363**	**0.367**
(2)	Interview different microphone in training and test	MFCC	2.99	0.562	0.633
		PLLR+Δ	6.39	0.804	0.819
		Fusion	**2.36**	**0.492**	**0.553**
(3)	Interview training telephone test	MFCC	3.63	0.625	**0.848**
		PLLR+Δ	9.20	0.862	0.978
		Fusion	**3.25**	**0.522**	0.874
(4)	Interview training telephone test rec. over microphone	MFCC	1.71	0.443	0.475
		PLLR+Δ	5.52	0.690	0.703
		Fusion	**1.69**	**0.372**	**0.406**
(5)	Telephone in training and test	MFCC	4.64	0.600	0.712
		PLLR+Δ	8.41	0.848	0.869
		Fusion	**4.29**	**0.560**	**0.688**

acoustic system (up to a 21% relative improvement in terms of EER), suggesting that PLLR features contain complementary information that can be used for SR.

TABLE 6.5: Results of i-vector /PLDA SR systems based on MFCC and PLLR features, and the fusion of them, on the NIST 2012 SRE core conditions 2 and 5.

Condition		System	EER	MinDCF	ActDCF
(2)	Telephone with No Added Noise	MFCC	1.77	0.272	0.290
		PLLR+Δ	3.12	0.419	0.440
		Fusion	**1.39**	**0.215**	**0.246**
(5)	Telephone Recorded in Noise	MFCC	1.93	0.260	0.294
		PLLR+Δ	3.72	0.449	0.481
		Fusion	**1.64**	**0.219**	**0.283**

6.6 Chapter Summary

In this chapter, the utility of PLLR features for SR tasks has been studied. As experiments for NIST 2010 and 2012 evaluation datasets have exhibited, the performance attained by systems based on PLLRs is far from the performance attained by similar systems based on other more common spectral features (such as MFCCs or PLP).

However, PLLR features do provide a way of extracting further information for SR tasks. The complementarity of PLLR features with regard to other spectral approaches has been empirically shown when fusing PLLR and MFCC based systems.

Chapter 7

Conclusions and future work

7.1 Conclusions

The research carried out to fulfill this thesis has followed a *natural* structure, as the conclusions attained after each study set the course of the following experimentation. The manuscript has been organized following that time-line structure, and the conclusions of each experimental chapter have been the unifying thread with the next one. The main conclusions of the work have been therefore introduced along the development of this manuscript.

The Phone Log-Likelihood Ratios have been formally defined. The former experimentation using the features integrated in a spoken language recognition i-vector system proved that the features provide an effective way to incorporate acoustic-phonetic information into frame-level features. PLLRs are easy to compute and to integrate in state-of-the-art language recognition systems, which makes them useful in different contexts and applications.

The effectiveness of these features was first tested in four different benchmarks, in which the systems making use of the features have consistently yielded competitive performances, outperforming, in most cases, those of the baseline acoustic (MFCC-SDC i-vector) and phonotactic (phone lattice-SVM) systems.

Pairwise fusions of the PLLR-based system with the baseline approaches showed that PLLRs provide complementary information to both, acoustic and phonotactic approaches. Furthermore, the fusion of the three systems still provides gains with

regard to the fusion of the baseline approaches, proving the complementarity of PLLR features with state-of-the-art features.

Next studies focused on reducing the dimensionality of feature vectors, so that common techniques applied on top of features like SDC could be also tested on PLLRs. Dimensionality reduction by means of supervised techniques, making use of information related to the phonetic categories in IPA charts, provided a way of decreasing the feature vector size into almost a third of the original size, with just a small system performance degradation. Among unsupervised techniques, PCA was successfully applied, showing that the set of features could be reduced to almost a third of their original size, not only without information loss, but actually enhancing system's performance.

The application of shifted delta transformation on top of the reduced set of PLLRs proved to be a convenient method to increase the potential of the features, and confirmed that (as previously found for MFCCs) using a larger spectro-temporal context expands the information conveyed the information carried out by the features.

The discovery that a reduced feature set, obtained by means of PCA, was more informative than the original led us to the analysis of the feature space. The research carried out revealed that the features were bounded by an hyper-surface, which was asymptotically tangential to the axes of the reference system. This would presumably limit the movement of the PLLR features with regard to the axes, limiting the information they provide. A projection method was developed to get rid of this effect, which projected the features to the top of the bounding surface, removing the asymptotic behavior. The system based on the projected set of features revealed that the method was a suitable transformation to be applied on the PLLRs. The combination of this technique with PCA further increased the effectiveness of the system.

Features were then tested on the noisy database RATS, using i-vector modeling approaches and logistic regression and neural networks as classifiers. This new series of experiments provided a new set of conditions to test the features, as the benchmark includes short time and noisy signals. Systems based on PLLR features attained high performance in all conditions.

Finally, the projected features were also tested in combination with the shifted delta transformation. This system, which combined the most significant improvements found in the presented studies, was tested on a NIST LRE benchmark and achieved the best result among all the experimentation carried out.

Research was extended to speaker recognition. The optimization of the PLLR extraction for a state-of-the-art i-vector PLDA approach revealed that similar configurations are optimal for both, SLR and SR systems, with an important difference: the

projection method does no improve the performance of the PLLR-based systems in SR tasks. Experimentation with the i-vector PLDA systems showed that the PLLR features are not as informative as the ones used in state-of-the-art systems, such as MFCCs. Nevertheless, when fusing PLLR-based and MFCC-based systems, there is still a gain in performance, revealing a complementarity between both approaches that could be useful to improve the overall system robustness.

7.2 Future Work

Extensive studies have been presented exploring the usefulness of PLLR features in different SLR benchmarks. Yet, the versatility of these features could be further analyzed, testing their merits in other systems and scenarios, which suggests several possible future research lines to give continuity to this project.

The approach has been widely tested under i-vector and Gaussian modeling techniques. Some systems have used PLLR-based i-vectors under LR or NN modeling. The PLLR features could still be tested under other modeling techniques, with the aim of finding the most advantageous modeling for PLLR based systems.

Most of the research conducted in this work has experimented on relatively long signals (30s). Results attained in RATS benchmarks for shorter signals (10s and 3s) suggest that PLLRs are suitable for recognition on short duration signals. This fact should be further checked in other benchmarks, like NIST LRE datasets.

Research could also focus on extracting PLLRs from other phone decoders, based on the phonetic inventory of other languages, or even trying to combine the outputs of different phone decoders.

Regarding speaker recognition, the fact that projected PLLR features do not enhance system performance opens an experimentation track. Studies should focus on trying to understand why the speaker-related information contained in the direction used for the projection cannot be properly modeled, or even whether (and why) speaker information is being lost after projecting PLLRs.

It would be interesting analyzing also the merits of PLLR features in Text Dependent Speaker Recognition (TDSR) tasks. The nature of this task, performing SR in utterances with fixed phonetic content, could be an optimal scenario to further understand the behavior of PLLR features for SR. Analysis of the performance of PLLR-based systems in TDSR tasks could help optimizing the speaker dependent information of the features, e.g. by discarding phonetic (utterance-dependent) content.

Besides the above mentioned research lines, Deep Neural Networks seem to be the way to go in SR and SLR fields. Latest works using DNNs have attained outstanding results in both tasks, outperforming other state-of-the-art techniques. In particular, given the success of PLLRs, which are typically extracted from the output layer of a neural network trained on phonetic classes (or states), a promising line of research involves extracting and using bottleneck features, or other kind of DNN-based features, at the frame level, just in the same way as it was done with PLLRs.

Appendix A

Datasets

This Appendix gives details about the database configuration used for the experiments presented in the manuscript, covering NIST 2007, 2009 and 2011 LRE, Albayzin 2010 LRE and NIST 2010 and 2012 SRE. In the case of RATS dataset, given that experiments were performed in collaboration with other sites, database partition details are referenced.

A.1 NIST 2007 LRE

The NIST 2007 LRE [96] defined a spoken language recognition task for conversational speech across telephone channels, involving 14 target languages.

Training and development data used in this thesis were limited to those distributed by NIST to all 2007 LRE participants: (1) the Call-Friend Corpus[1]; (2) the OHSU Corpus provided by NIST for the 2005 LRE[2]; and (3) the development corpus provided by NIST for the 2007 LRE[3]. A set of 23 languages/dialects was defined for training, including target and non-target[4] languages. For development purposes, 10 conversations per language were randomly selected, and the remaining conversations (amounting to around 968 hours) were used for training. Development conversations were further divided into 30-second speech segments. The total number of 30-second segments was 3073 (see Table A.1 for more details). Results reported in this thesis

[1] See http://www.ldc.upenn.edu/.
[2] OHSU Corpora, http://www.ohsu.edu/.
[3] See http://www.itl.nist.gov/iad/mig/tests/lre/2007/.
[4] French was the only non-target language used for NIST 2007 LRE.

have been computed on the subset of 30-second speech segments of the test set for the closed-set condition (2158 segments), which was the primary task in the NIST 2007 LRE.

TABLE A.1: 2007 NIST LRE core condition: training data (hours), development and evaluation data (# 30s segments), disaggregated for target and non-target languages.

	Hours	**# 30s cuts**	
Language	Train	Devel	Eval
Arabic	52.59	179	80
Bengali	5.0	76	80
Chinese	166.12	567	398
English	143.7	288	240
Farsi	46.22	225	80
German	57.03	173	80
Industani	64.35	243	240
Japanese	79.11	141	80
Korean	72.86	150	80
Russian	5.0	66	160
Spanish	117.35	531	240
Tamil	58.18	165	160
Thai	5.0	64	80
Vietnamese	46.7	205	160
Non-Target	48.74	-	-
TOTAL	967.95	3073	2158

A.2 NIST 2009 LRE

The NIST 2009 LRE featured 23 target languages [93], involving 11 target languages for which Conversational Telephone Speech (CTS) was available in the NIST 2007 LRE dataset, plus 12 target languages not seen in previous NIST LRE for which Broadcast Narrow-Band Speech was provided. Most of the speech provided for the latter consisted of telephone calls included in Voice of America (VOA) broadcasts. For the 12 new target languages, NIST distributed between 141 and 199 30-second audited VOA segments per language. Additional non-audited materials were provided for the 23 target languages and for several non-target languages (see Table A.2).

Training and development data used in this thesis were limited to those distributed by NIST to all 2009 LRE participants. A set of 64 languages/dialects was defined

TABLE A.2: 2009 NIST LRE core condition: training data (hours), development and evaluation data (# 30s segments), disaggregated for target and non-target languages.

	Hours		**# 30s cuts**			
	Train		Devel			Eval
Language	2007 (CTS)	2009 (VOA)	2007 (CTS) devel	2007 (CTS) eval	2009 (VOA)	2009
Amharic	-	58.31	-	-	262	398
Bosnian	-	5.63	-	-	259	355
Cantonese	5.0	2.45	83	80	104	378
Creole	-	7.21	-	-	256	323
Croatian	-	6.45	-	-	190	376
Dari	-	69.05	-	-	276	389
EngAmerican	130.7	9.01	204	80	230	896
EngIndian	13.0	-	84	160	-	574
Farsi/Persian	46.22	25.16	225	80	294	390
French	48.74	67.91	222	80	293	395
Georgian	-	4.32	-	-	166	399
Hausa	-	48.31	-	-	274	389
Hindi	59.35	10.06	174	160	178	667
Korean	72.86	5.70	150	80	250	463
Mandarin	151.1	32.4	331	158	230	1015
Pashto	-	184.3	-	-	281	395
Portuguese	-	25.74	-	-	240	397
Russian	5.0	147.76	66	160	299	511
Spanish	117.35	45.44	531	240	242	385
Turkish	-	6.67	-	-	289	394
Ukrainian	-	5.59	-	-	281	388
Urdu	5.0	36.60	69	80	299	379
Vietnamese	46.7	9.5	205	160	240	315
Non-Target	266.92	408.82	-	-	-	-
TOTAL	967.95	1222.39	2344	1518	5433	10571

for training models. Each of them was mapped either to a target language or to non-target languages[5]. For example, Mainland and Taiwan Chinese from NIST 2007 LRE and Mandarin Chinese from VOA were all mapped to Mandarin Chinese, whereas Arabic was mapped to non-target languages. Persian and Farsi were mapped to the same language, as was properly pointed out in [77].

For languages appearing in VOA recordings, the longest speech segments out of each file were posted to the training dataset, using no more than 2 segments per file, and a minimum of 225 segments per language. The number of segments extracted per file was relaxed (augmented) for those languages with few files in VOA.

The whole training dataset (CTS from NIST 2007 LRE and VOA broadcast speech from NIST 2009 LRE) amounted to 2190 hours. For development, some materials taken from the development and evaluation datasets of the NIST 2007 LRE were

[5]The set of non-target languages defined for the NIST 2009 LRE included: Arabic, Bengali, German, Japanese, Tamil and Thai from CTS recordings, and Albanian, Azerbaijani, Bangla, Burmese, Greek, Indonesian, Khmer, Kinyarwanda/Kirundi, Kurdish, Macedonian, Ndebele, Oromo, Serbian, Shona, Somali, Swahili, Tibetan, Tigrigna, and Uzbek from VOA broadcasts.

TABLE A.3: NIST 2011 LRE core condition: training data (hours) disaggregated for target and non-target languages.

Language	Hours			
	Train			
	2007 (CTS)	2009 (VOA)	2011 (30s audit)	Other sources
Arabic Iraqi	-	-	0.48	20.34
Arabic Levantine	-	-	0.47	27.56
Arabic Maghrebi	-	-	0.41	1.79
Arabic MSA	-	-	0.47	1.87
Bengali	5.0	54.40	-	-
Czech	-	-	0.41	4.19
Dari	-	69.05	-	-
English American	130.7	9.01	-	-
English Indian	13.0	-	-	-
Farsi/Persian	46.22	25.16	-	-
Hindi	59.35	10.06	-	-
Lao	-	-	0.50	2.22
Mandarin	151.1	32.40	-	-
Panjabi	-	-	0.50	-
Pashto	-	184.38	-	-
Polish	-	-	0.51	1.79
Russian	5.0	147.76	-	-
Slovak	-	-	0.41	1.69
Spanish	117.3	45.44	-	-
Tamil	58.18	-	-	-
Thai	5.0	-	-	-
Turkish	-	6.67	-	-
Ukrainian	-	5.59	-	-
Urdu	5.0	36.60	-	-
Non-Target	257.76	406.93	-	-
TOTAL	853.61	1033.45	4.16	61.45

used (see Table A.1). For languages appearing in VOA, besides the audited segments provided by NIST, additional randomly extracted speech segments, each around 30 seconds long (specifically, between 25 and 35 seconds long), were used. The whole development dataset consisted of 9295 segments. Results reported in this thesis were computed on the NIST 2009 LRE evaluation corpus, specifically on the 30-second, closed-set condition (primary evaluation task).

A.3 NIST 2011 LRE

In the NIST 2011 LRE, 24 target languages were considered (see Tables A.3, A.4). Among them, 9 languages had never been used before in NIST LRE. Development data specifically collected for these 9 languages were sent to participants, including 100 30-second segments per language. For a better coverage, these subsets were randomly split into two disjoint subsets (each having approximately half the segments for each language/dialect): the first half was used to train specific models for

TABLE A.4: NIST 2011 LRE core condition: development and evaluation data (30s segments), disaggregated for target and non-target languages.

	# 30s cuts					
	Devel					Eval
	2007		2009		2011	2011
Language	(CTS)	(eval)	(VOA)	(eval)	(audit)	
Arabic Iraqi	-	-	-	-	48	308
Arabic Levantine	-	-	-	-	49	308
Arabic Maghrebi	-	-	-	-	54	305
Arabic MSA	-	-	-	-	51	306
Bengali	76	80	296	43	-	412
Czech	-	-	-	-	56	261
Dari	-	-	276	389	-	267
English American	204	80	230	896	-	221
English Indian	84	160	-	574	-	387
Farsi/Persian	225	80	294	390	-	404
Hindi	174	160	178	667	-	213
Lao	-	-	-	-	41	62
Mandarin	331	158	230	1015	-	360
Panjabi	-	32	-	9	45	299
Pashto	-	-	281	395	-	383
Polish	-	-	-	-	46	267
Russian	66	160	299	511	-	441
Slovak	-	-	-	-	56	280
Spanish	531	240	242	385	-	419
Tamil	165	160	-	-	-	414
Thai	64	80	-	188	-	375
Turkish	-	-	289	394	-	276
Ukrainian	-	-	281	388	-	170
Urdu	69	80	299	379	-	478
Non-Target	-	-	-	-	-	-
TOTAL	1989	1470	3135	6623	446	7616

the new languages, and the second half was used to estimate backend and fusion parameters [111].

To train more robust models for the target languages, additional data was included from databases distributed by the Linguistic Data Consortium (LDC), some of them containing conversational telephone speech (LDC2006S45 for Arabic Iraqi, LDC2006S29 for Arabic Levantine) and others containing broadcast speech (LDC2000S89 and LDC2009S02 for Czech). For these latter, only automatically detected telephone-speech segments were used.

The remaining materials were extracted from wide-band broadcast news recordings, dowsampling them to 8 kHz and applying the Filtering and Noise Adding Tool[6] (FANT) to simulate a telephone channel. The COST278 Broadcast News database [151] was used to get speech segments for Czech and Slovak. Arabic MSA was extracted from Al Jazeera broadcasts included in the KALAKA-2 database created for the Albayzin 2010 LRE [128]. Finally, broadcasts were also *captured* from video

[6]Available online: http://dnt.kr.hsnr.de

archives in TV websites to get speech segments in Arabic Maghrebi (Arrabia TV, http://www.arrabia.ma) and Polish (Telewizja Polska, TVP INFO, http://tvp.info). TV broadcasts were fully audited, so that only reasonably clean speech segments were selected for training. It was not feasible to collect additional training materials for Panjabi by any means. Therefore, a single model (trained on just 55 segments) was used for this language.

A set of 66 languages/dialects was defined for training. Each of them was mapped either to a target language or to non-target languages[7]. The training dataset included the data mentioned above (one-half of the audited segments plus other sources) plus 2007 CTS and 2009 VOA signals (see Tables A.1 and A.2). The whole training dataset for the NIST 2011 LRE benchmark amounted to 1953 hours.

For development purposes, the second half of the audited segments provided for new target languages, along with the NIST 2007 and 2009 evaluation datasets, and 30-second signals used for development in 2007 and 2009 (see Tables A.1 and A.2) were used. The whole development dataset consisted of 13663 segments. Results reported in this paper were computed on the NIST 2011 LRE evaluation corpus, specifically on the 30-second, closed-set condition (primary evaluation task) (see Table A.4 for more details).

A.4 Albayzin 2010 LRE (KALAKA-2)

The Albayzin 2010 LRE dataset (KALAKA-2) included wide-band 16 kHz TV broadcast speech signals for six target languages (see Table A.5). The Albayzin 2010 LRE [128] featured two main evaluation tasks, on clean and noisy speech, respectively. In this thesis, acoustic processing involved downsampling signals to 8 kHz, since all the systems were designed to deal with narrow-band signals.

The training, development and evaluation datasets used for this benchmark matched exactly those defined for the Albayzin 2010 LRE. For the primary clean-speech language recognition task, more than 10 hours of clean speech per target language were used for training. For the noisy-speech language recognition task, besides the clean speech subset, more than 2 hours of noisy/overlapped speech segments were used for each target language. The distribution of training data, which amounts to around 82 hours, is shown in Table A.5. Only 30-second segments were used for development purposes. The development dataset used in this thesis consisted

[7]The set of non-target languages defined for the NIST 2011 LRE included: French, German, Japanese, Korean and Vietnamese from CTS recordings, and Albanian, Amharic, Creole, French, Georgian, Greek, Hausa, Indonesian, Kinyarwanda/Kirundi, Korean, Ndebele, Oromo, Shona, Somali, Swahili, Tibetan and Tigrigna from VOA broadcasts.

of 1192 segments, amounting to more than 10 hours of speech. Results reported in this thesis were computed on the 30-second, closed set condition (for both clean speech and noisy speech conditions) of the Albayzin 2010 LRE evaluation corpus. The distribution of segments in the development and evaluation datasets is shown in Table A.5. For further details, see [129].

TABLE A.5: Albayzin 2010 LRE: Distribution of training data (hours) and development and evaluation data (30s segments).

	Clean Speech			Noisy Speech		
	Hours	# 30s cuts		Hours	# 30s cuts	
Language	Train	Devel	Eval	Train	Devel	Eval
Basque	10.73	146	130	2.25	29	74
Catalan	11.45	120	149	2.18	47	55
English	12.18	133	135	2.53	60	69
Galician	10.74	137	121	2.23	60	83
Portuguese	11.08	164	146	3.28	77	58
Spanish	10.41	136	125	3.70	83	79
TOTAL	66.59	836	806	16.17	356	418

A.5 NIST 2010 SRE

Table A.6 shows the speaker distribution across different NIST SRE datasets. The elements in the diagonal show the number of speakers per dataset and the ones outside the diagonal represent the number of speakers shared by the corresponding pair of datasets.

TABLE A.6: Number of speakers uniquely included in each dataset (diagonal) and shared with other datasets (outside the diagonal).

	SRE04	SRE05	SRE06	SRE08	FU08
SRE04	310	0	0	0	0
SRE05	0	525	348	0	0
SRE06	0	348	949	112	0
SRE08	0	0	112	1336	150
FU08	0	0	0	150	150

These datasets were used to support the development of different parts of the system. The partition was performed as follows:

• NIST 2004, 2005 and 2006 SRE: These datasets were used to obtain the training signals for the Universal Background Model, Channel Compensation, Impostor, ZNorm and TNorm sets. Table A.7 displays the number of signals assigned to each set.

Table A.7: NIST 2004, 2005 and 2006 SRE signal distribution.

	Women	Men	Total
UBM	2804	2119	4923
CHC	4586	3531	8117
IMP	2780	2094	4874
TNorm	1479	960	2439
ZNorm	1403	1146	2549

• NIST 2008 SRE: Datasets for this series of experiments were defined so that they included disjoint sets of speakers, thus, all the signals of NIST 2008 SRE belonging to the 112 speakers present in previous databases were discarded. The rest of the signals were divided into two groups: dev1 and dev2. Table A.8 shows the number of signals in each of these subsets.

Table A.8: NIST 2008 SRE signal distribution.

	SRE08_reduced	**dev1**	**dev2**
training	3149	1621	1528
test	6211	3306	2905

• NIST 2008 Follow Up: Some of the signals present in this dataset were recorded with the same microphone types used for NIST 2004-2008 datasets, whereas others were recorded using new microphone types. To take advantage of this particularity, that could provide robustness against channel variabilities, signals were divided into three subsets: Channel compensation, ZNorm and TNorm. Table A.9 shows the number of signals included in each subset.

Table A.9: NIST 2008 Follow Up signal distribution.

	Speakers		Signals		
	Women	Men	Women	Men	All
CHC	38	38	2432	1776	4208
TNorm	18	18	1145	848	1993
ZNorm	19	19	1212	875	2087

The dataset used for UBM training is a subset of NIST 2004, 2005 and 2006 SREs consisting of 4882 speech files, as defined in [108]. The approaches presented in [46, 106, 108] used another subset of signals from the same datasets, that amounted to 8117 speech files, for channel compensation.

For the systems using i-vector-PLDA approaches, the Total Variability matrices were trained on the channel compensation subset (including 8117 signals) defined previously. The PLDA models were trained on a subset of the NIST 2004, 2005,

2006 and 2008 SRE datasets plus NIST 2008 *Follow-up* signals, created with the signals from the IMP, TNorm, ZNorm and NIST 2008 SRE reduced corpora, which amounted to 23302 speech files.

A.6 NIST 2012 SRE

The dataset used for UBM training is the same defined for the NIST 2010 SRE benchmark: a subset of SRE04, SRE05 and SRE06 consisting of 4882 speech files.

For the NIST 2012 SRE experiments, the Total Variability matrices and the PLDA models shared a common training set, including speakers from the NIST 2006, 2008 and 2010 SRE datasets and avoiding repeated speech. This file list was the *single_file_per_ldcid_map* provided by NIST. Besides those segments, 590 channel-balanced randomly chosen signals from the NIST 2008 *Follow-Up* set were used and the 100 *single utterance per speaker* signals provided for the NIST 2012 SRE. The whole training set amounted to 21176 speech files, as described in [49].

TABLE A.10: NIST 2012 SRE iVector and PLDA training signal distribution.

	Speakers		Signals		
	Women	Men	Women	Men	All
NIST 2006 SRE	72	38	754	434	1188
NIST 2008 SRE	799	461	7693	4419	12112
NIST 2010 SRE	260	240	3746	3240	7186
NIST 2008 Follow Up	86	63	339	251	590
NIST 2012 SRE Single_spk	60	40	60	40	100

A.7 RATS

Experiments on RATS dataset [117], were performed using the data configuration and partitions as defined in [99, 116].

Appendix B

Participation in International Challenges

The work carried out comprised also building several systems which were part of the submissions of the research group (GTTS, http://gtts.ehu.es) to several international evaluations. This Appendix provides details about the participation in some of them: a short description of the evaluation, specifications of the submitted systems, the development stages in which the author contributed and the attained results.

NIST evaluation rules prevent participants from commenting on the rank their systems have attained or sharing results from other participants. Therefore, for those challenges, this section will only provide system development and official results for our site, and comments will only be made comparing the different approaches presented by our group.

B.1 NIST SRE 2010

Evaluation:

The NIST 2010 SRE followed the spirit of previous evaluations (see 6.1.1 for details). The evaluation focused on both, telephone and microphone speech, recorded over different types of channels. Further details can be found in [4, 94].

System:

The EHU[1] system was built by discriminatively fusing four subsystems: a GMM-SVM sub-system, a Linearized Eigenchannel GMM (LE-GMM) subsystem, a GLDS-SVM subsystem and a JFA subsystem.

The Qualcomm-ICSI-OGI (QIO) noise reduction technique (based on Wiener filtering) was independently applied to the audio streams. The full audio stream was taken as input to estimate noise characteristics.

Features were obtained with the Sautrela toolkit[2] [105]. Mel-Frequency Cepstral Coefficients (MFCC) were used as acoustic features, computed as described in Section 6.3. Two gender dependent UBMs consisting of 1024 mixture components were trained with the Sautrela toolkit.

- GMM-SVM & LE-GMM subsystems: The GMM-SVM and LE-GMM (also known as dot-scoring) subsystems were built following the SUNSDV system description for NIST 2008 SRE [145]. Channel compensation was trained for inter-telephone, inter-microphone and telephone-microphone variations, using 20, 20 and 40 eigenchannels, respectively. For the GMM-SVM subsystem, a linear kernel was trained using SMVTorch [36].
- GLDS-SVM subsystem: Sufficient statistics space compensation was projected to feature space by applying the following expression:

$$\hat{\mathbf{f}}_t = \mathbf{f}_t - \sum_k \frac{\gamma_k(t)}{n_k} \boldsymbol{\Sigma}_k^{\frac{1}{2}} c_k^S \tag{B.1}$$

  where $\mathbf{f}_t$ is the feature vector at time t, $\gamma_k(t)$ is the posterior of Gaussian k at time t, $n_k = \sum_t \gamma_k(t)$ is the zero-order statistic of Gaussian k, $\boldsymbol{\Sigma}_k$ is the diagonal covariance matrix of Gaussian k and c_k^S is the first-order statistics shift (sufficient statistics space compensation factor) of Gaussian k given the input segment S. A polynomial expansion of degree 3 and a Generalized Linear Discriminant Sequence Kernel [28] were then applied.
- JFA subsystem: The Joint Factor Analysis Matlab Demo from BUT [1, 83] was applied to the MFCC + Δ + $\Delta\Delta$ features, using 200 eigenvoices and 100 eigenchannels.

Trials were conditioned on three channel types: no microphone sessions (0MIC), one microphone session (1MIC) and two microphone sessions (2MIC). Gender dependent

[1] In NIST evaluation submissions EHU is used as acronym for our systems
[2] http://gtts.ehu.es/TWiki/bin/view/Sautrela

and channel type condition dependent ZT normalization was performed on trial scores.

Side-info-conditional fusion and calibration was performed with FoCal [62], using channel type and gender conditioning. Fused scores were calibrated to be interpreted as detection log-likelihood-ratios, and a Bayes threshold of 6.907 was applied to make the hard decisions.

Contribution:

- Development of both GMM-SVM subsystems.
- Sufficient statistic compensation.
- Development of JFA subsystem.

Results:

GMM-SVM and dot-scoring approaches stood out as the best performing individual systems, followed by the GLDS-SVM and JFA systems. The fusion of the systems attained the best results, not being yet significantly better than the best individual systems.

Figure B.1 shows the DET curve of the fused system for the main evaluation conditions. As shown in the Figure, there was a noticeable calibration error in the condition tests involving microphone signals (conditions 1, 2 and 4). On the other hand, systems were well calibrated for test conditions related to telephone speech.

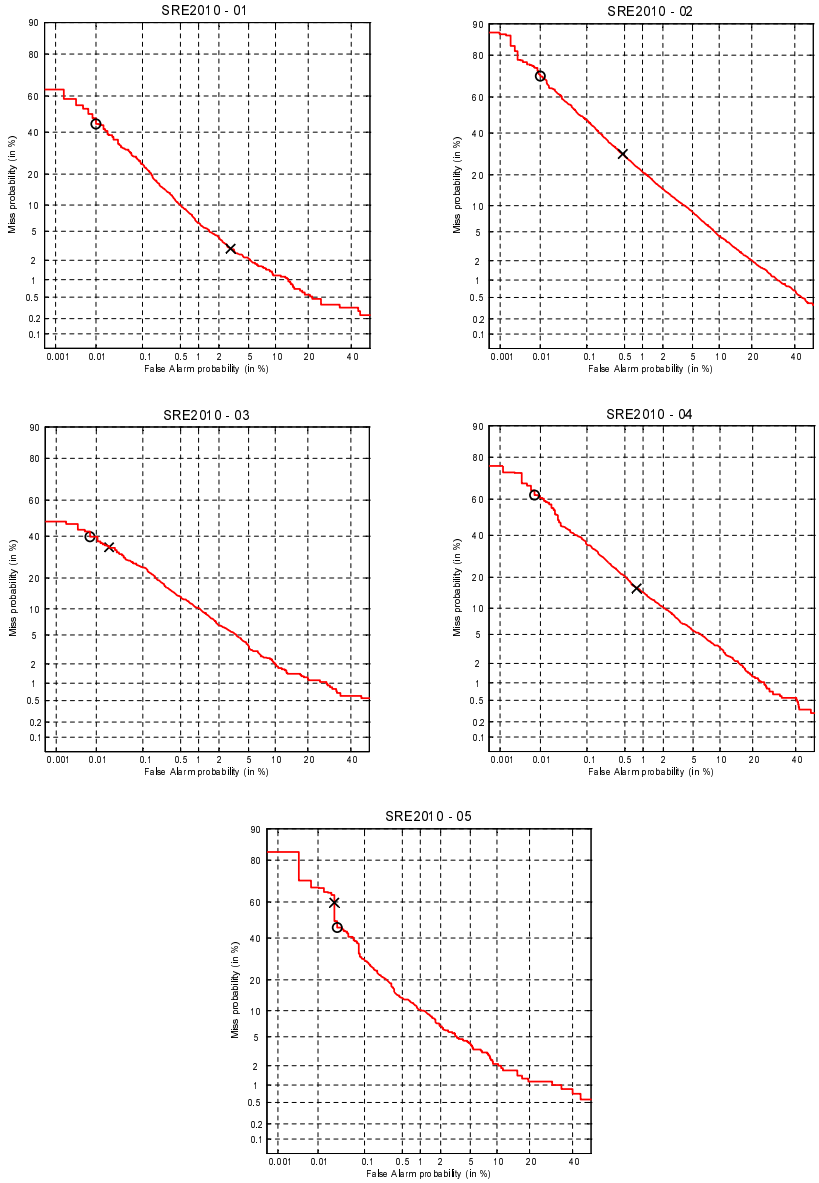


FIGURE B.1: DET curves of the EHU fused system for the NIST 2010 SRE core test conditions.

B.2 NIST LRE 2011

Evaluation: In the challenge organized in 2011, the main task differed from previous NIST LRE evaluations, as it focused on language pair recognition. The evaluation comprised 24 target languages, 9 of which had never been used before. For more details, see the evaluation plan [8] and the paper reporting the evaluation analysis [69].

System: 5 subsystems were trained for the evaluation, which were then fused using four different configurations, to build one primary and three contrastive systems. The five mentioned subsystems were designed as follows:

- Three high-level (phonotactic) phone-lattice SVM subsystems, based on the BUT decoders for Czech, Hungarian and Russian.

  The three subsystems followed the same configuration. First, energy based VAD was applied to remove non-speech segments. BUT TRAPS/NN CZ, HU and RU phone decoders were used to perform phone tokenization. The three non-phonetic units (*int, short, pau*) were integrated into a single non-phonetic unit. Phone state posteriors and phone-lattices were computed by means of HTK[3] [156]. The lattice-tool from SRILM[4] [144] was used to estimate phone n-grams. LIBLINEAR[5] [59] was used to train SVM classifiers, where SVM vectors consisted of expected counts of n-grams extracted from the lattices, weighted by their background probabilities.

- Two low-level (acoustic) subsystems: a Linearized Eigenchannel GMM subsystem based on channel compensated statistics and Generative i-vector subsystem computed from channel compensated statistics.

  Both subsystems relied on MFCC-SDC features, computed under a 7-2-3-7 configuration. A gender independent 1024-mixture GMM-UBM was trained by EM-ML using binary mixture splitting, orphan mixture discarding and variance flooring. For each input utterance, UBM-MAP adaptation was applied and zero and first order statistics were computed and used as features.

  – *Dot-Scoring (LE-GMM) system:* Channel compensation was performed as outlined in *Eigenchannel Compensation* in Section 2.4. The scoring was computed in the following way:

$$score(S,l) = log\frac{P(S|\lambda_l)}{P(S|\lambda_{\text{ubm}})} = \hat{\mathbf{m}}_l^t \cdot \hat{\mathbf{x}}_S \tag{B.2}$$

[3] http://htk.eng.cam.ac.uk/docs/docs.shtml
[4] http://www.speech.sri.com/projects/srilm/
[5] http://www.csie.ntu.edu.tw/∼cjlin/liblinear

where λ_l and λ_{ubm} are the target language model and the UBM model, respectively, $\hat{\mathbf{x}}_S$ are the channel compensated first order statistics of the target signal S, and $\hat{\mathbf{m}}_l$ are the centered and normalized channel compensated MAP-means of language l, computed according to Equation 2.23:

$$\hat{\mathbf{m}}_l = (\tau\mathbf{I} + diag(\mathbf{n}_l))^{-1}\hat{\mathbf{x}}_l \tag{B.3}$$

where $\tau = 16$ is the relevance factor, $\mathbf{n}_l$ are the zero order statistics of language l, and $\hat{\mathbf{x}}_l$ are the channel compensated first order statistics of the language model.

– *i-vector system:* An i-vector approach, as defined in *Total Variability Factor Analysis* in Section 2.4, was used, in which 500 dimensional i-vectors were computed from channel compensated sufficient statistics using only training data from target languages. A generative Gaussian approach was used for scoring (each language being modeled by a single Gaussian).

Subsystems were combined in four different ways, depending on the normalization applied, and the data used for training the backend and fusion parameters. Details of these four configurations are summarized in Table B.1:

TABLE B.1: Backend and fusion configuration for the EHU systems submitted to the NIST 2011 LRE.

System	*zt-norm*	Backend and Fusion Training dataset		
		30s	10s	3s
Pri	No	dev30	dev10	dev03
Con1	No	dev30	dev10+dev30	dev03+dev10+dev30
Con2	Yes	dev30	dev10	dev03
Con3	Yes	dev30	dev10+dev30	dev03+dev10+dev30

Contribution:

- Development of both acoustic subsystems.
- System calibration and fusion tunning.

Results:

Table B.2 shows the performance of all individual systems submitted on the 30s test set. All phonotactic and acoustic individual systems performed similarly. The

fusion of three phonotactic systems outperformed the fusion of the acoustic approaches. The fusion of all 5 systems attained still a significant gain, achieving the best performance overall.

TABLE B.2: Performance (in terms of C_{avg}) of the phonotactic and acoustic subsystems and partial and complete fusions on the NIST 2011 LRE 30s test set.

	C^{24}_{avg}		C_{avg}	
	min	act	min	act
Phone-CZ	12.15	14.02	2.97	3.76
Phone-HU	11.96	14.28	2.71	3.62
Phone-RU	11.38	13.76	2.57	3.46
Phonotactic	7.73	10.13	1.47	2.28
Dot-Scoring	11.62	14.18	2.19	3.17
i-vector	11.58	14.15	2.60	3.50
Acoustic	11.18	13.30	2.00	2.85
All	6.15	8.95	0.93	1.69

Table B.3 shows the results of the fused system in all test sets (30s, 10s and 3s) as well as the results for the contrastive systems. No significant differences can be found between their performances except on the 3s condition, where contrastive system 3 yielded slightly better figures. Further details can be found in [113] and [114].

TABLE B.3: Official NIST 2011 LRE results for the EHU systems.

30s	min C_{avg}^{24}	act C_{avg}^{24}	min C_{avg}	act C_{avg}
Pri	6.15	8.95	0.93	1.69
Con1	6.15	8.95	0.94	1.69
Con2	6.08	9.09	0.91	1.75
Con3	6.07	9.07	0.91	1.75
10s	min C_{avg}^{24}	act C_{avg}^{24}	min C_{avg}	act C_{avg}
Pri	12.99	14.77	3.37	4.08
Con1	12.44	14.55	3.23	4.03
Con2	12.72	14.68	3.31	4.12
Con3	12.36	14.36	3.14	3.95
3s	min C_{avg}^{24}	act C_{avg}^{24}	min C_{avg}	act C_{avg}
Pri	25.54	27.25	11.60	12.88
Con1	23.97	25.34	11.07	12.05
Con2	25.52	27.05	11.62	12.86
Con3	23.31	25.28	10.87	12.06

B.3 NIST SRE 2012

Evaluation:

NIST 2012 SRE focused, as previous evaluations, on speaker detection tasks, but included new conditions and challenges making the evaluation different from previous ones.

NIST 2012 SRE included noisy speech in some test segments, making the detection task more demanding. The evaluation task also included tracks with distinctions regarding known/unknown trials. Besides, or the first time in NIST SRE, the use of information from other target trials was allowed to compute the trial score. For more details on the evaluation, see [5, 70].

System:

The primary system submitted by EHU consisted of the fusion of two subsystems. Both subsystems were based on an i-vector PLDA approach (see *Total Variability Factor Analysis* in Section 2.4 and Section 6.1.3 for details), using MFCC and PLLR features, respectively.

MFCC features were computed following the procedure described in 6.3 resulting in a 39-dimensional feature vector. PLLRs were computed using the BUT TRAPs/NN phone decoder for Hungarian, as shown in Section 3.3, producing a 59-dimensional feature vector at each frame t. Voice activity detection was performed by removing the feature vectors whose highest PLLR value corresponded to the integrated non-phonetic unit.

Training and evaluation sets were defined as described in A.6. The Qualcomm-ICSI-OGI (QIO)[10] noise reduction technique was independently applied to the audio streams. The full audio stream was taken as input to estimate noise characteristics.

Two gender dependent UBMs, each consisting of 1024 mixture components, were estimated on the same dataset used for the EHU NIST 2010 SRE submission (see A.6), using the Sautrela toolkit. Two gender dependent Total Variability matrices were estimated on the whole training set, by means of Sautrela. The i-vector dimensionality was set to 500. Gaussian PLDA was also estimated on the whole training set [18].

PLDA system scores $s(\mathcal{T}, S)$ were used as log-likelihoods, so that the likelihood of a test utterance S given a speaker i was computed as the average likelihood of S over

all the training signals $\mathcal{T}$ of that speaker, as follows:

$$p(S|i) = \frac{1}{|Train(i)|} \sum_{\mathcal{T} \in Train(i)} e^{s(\mathcal{T},S)} \tag{B.4}$$

Finally, speaker log-likelihood ratios were computed from speaker log-likelihoods using flat priors.

Calibration and fusion were estimated and applied by means of the Bosaris toolkit [19], using the whole training set to estimate calibration/fusion parameters.

Contribution:

- Development of both subsystems.
- System calibration and fusion tunning.

Results:

Results on the SRE 2012 dataset are shown in Table B.4. Note that no special treatment for noisy conditions was performed to obtain these results. Once again, the result attained by the acoustic system is better that the one obtained with the PLLR-based approach in all conditions. However, the fusion of both approaches attains up to a 38% relative improvement in terms of MinDCF and up to a 29% relative improvement in terms of ActDCF with regard to the acoustic approach, which reveals a complementarity between the features.

TABLE B.4: Results of EHU i-vector-PLDA systems based on MFCC and PLLR features, and the fusion of them, on the NIST 2012 SRE core conditions.

Condition	System	EER	MinDCF	ActDCF
Interview with No Added Noise	MFCC	9.00	0.514	0.716
	PLLR + Δ	13.85	0.620	0.751
	Fusion	9.38	0.486	0.548
Telephone with No Added Noise	MFCC	1.83	0.277	0.296
	PLLR + Δ	3.12	0.419	0.440
	Fusion	1.39	0.213	0.239
Interview with Added Noise	MFCC	9.98	0.533	1.024
	PLLR + Δ	14.98	0.615	0.842
	Fusion	10.57	0.514	0.726
Telephone with Added Noise	MFCC	7.05	0.510	0.519
	PLLR + Δ	7.73	0.611	0.619
	Fusion	5.60	0.430	0.465
Telephone Recorded in Noise	MFCC	2.11	0.312	0.404
	PLLR + Δ	3.72	0.449	0.481
	Fusion	1.69	0.198	0.562

B.4 MOBIO 2013

Evaluation:

MOBIO (MObile BIOmetry) is a face and speaker database containing speech utterances (and videos) of 52 female and 100 male speakers. All the data in this benchmark was collected using mobile devices [101]. The audio signals comprise both planned and spontaneous speech.

The MOBIO 2013 evaluation proposed a speaker recognition task in a highly demanding benchmark, with signals extracted from telephone conversations obtained with mobile devices, that contained audio segments of short durations and speech in noisy conditions.

System:

The EHU system was based on a i-vector-PLDA approach (see *Total Variability Factor Analysis* in Section 2.4 and Section 6.1.3 for details). The feature extraction and the VAD were done using the Sautrela toolkit [105]. PLDA [100] was applied directly on the extracted 500 dimensional i-vector space. A gender independent 1024 component UBM, an i-vector extractor, and gender dependent PLDA systems were trained on the background set of the MOBIO database. The development dataset was used only to estimate the calibration parameters. Also, a collaboration was made with *Laboratorio de sistemas de Língua Falada* (L2F), fusing the systems submitted by both groups to the MOBIO evaluation.

Contribution:

- System development.
- System Calibration and fusion tunning.

Results:

Table B.5 shows the EER of the systems submitted by EHU and L2F, as well as the best performing primary systems submitted to the competition (considering the evaluation set). All EHU systems performed significantly better on the male test set than on the female test set. The performance of the L2F-EHU fused system was comparable to that of the most competitive systems. The fusion of all the systems (11 in total) attained a significant gain with regard to any of the individual approaches. Figure B.2 shows the DET curves for all the submitted systems on the female and male test sets.

TABLE B.5: Official MOBIO 2013 results for several primary systems.

System	Female EER	Male EER
Alpineon	10.678	7.076
GIAPSI	12.813	8.865
Mines-Telecom	11.633	9.109
EHU	19.511	10.058
L2F	22.140	11.129
L2F-EHU	17.266	8.191
Fusion (11 systems)	6.986	4.767

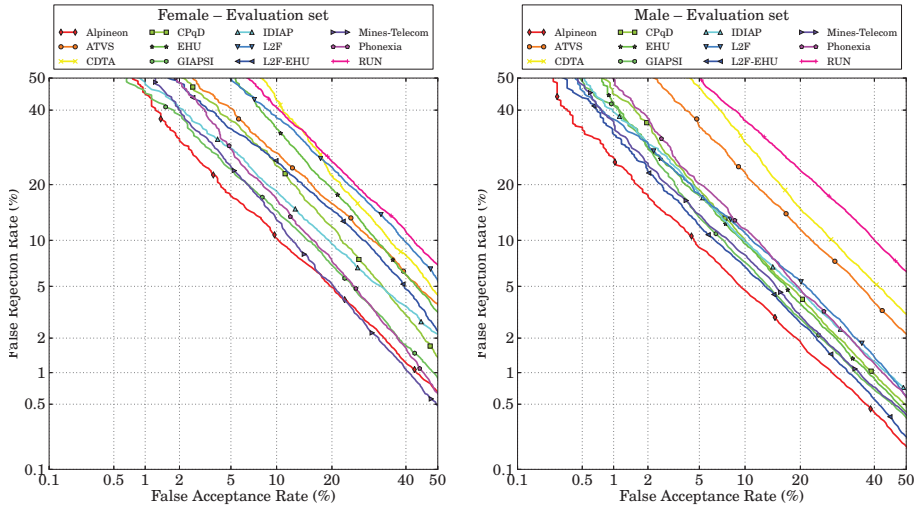


FIGURE B.2: DET curves of the primary systems submitted to MOBIO 2013 (images taken from [84])

Bibliography

[1] *Joint Factor Analysis Matlab Demo.* http://speech.fit.vutbr.cz/en/software/jointfactor-analysis-matlab-demo.

[2] *Linguistic Data Consortium.* https://www.ldc.upenn.edu/.

[3] *NIST LRE.* http://www.itl.nist.gov/iad/mig/tests/lre/.

[4] *The NIST Year 2010 Speaker Recognition Evaluation Plan.* http://www.itl.nist.gov/iad/mig/tests/spk/2010/index.html.

[5] *The NIST Year 2012 Speaker Recognition Evaluation Plan.* http://www.nist.gov/itl/iad/mig/upload/NIST_SRE12_evalplan-v17-r1.pdf.

[6] *PLLR computation software.* https://sites.google.com/site/gttspllrfeatures/home.

[7] *Software Technologies Working Group.* Department of Electricity and Electronics, Faculty of Science and Technology, University of the Basque Country. http://gtts.ehu.es.

[8] *The 2011 NIST Language Recognition Evaluation Plan (LRE11).* http://www.nist.gov/itl/iad/mig/upload/LRE11_EvalPlan_releasev1.pdf.

[9] *IPA Chart*, 2005. http://www.langsci.ucl.ac.uk/ipa/ipachart.html.

[10] A. Adami, L. Burget, S. Dupont, H. Garudadri, F. Grezl, H. Hermansky, P. Jain, S. Kajarekar, N. Morgan, and S. Sivadas. Qualcomm-ICSI-OGI features for ASR. In *Proceedings of ICSLP2002*, 2002.

[11] A. G. Adami. Modeling prosodic differences for speaker recognition. *Speech Communication*, 49(4):277 – 291, 2007.

[12] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas. Score normalization for text-independent speaker verification systems. *Digital Signal Processing*, 10(1-3):42–54, January 2000.

[13] J. Benesty, M. M. Sondhi, and Y. Huang, editors. *Springer Handbook of Speech Processing*. Springer, 2008.

[14] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[15] N. Brümmer. *Measuring, refining and calibrating speaker and language information extracted from speech*. PhD thesis, Department of Electrical and Electronic Engineering, University of Stellenbosch, Private Bag X1, 7602 Matieland, South Africa, 2010.

[16] N. Brummer. The EM algorithm and Minimum Divergence applied to PLDA. Technical report, 2010.

[17] N. Brümmer, L. Burget, J. Cernocky, O. Glembek, F. Grezl, M. Karafiat, D. van Leeuwen, P. Matejka, P. Schwarz, and A. Strasheim. Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006. *IEEE Transactions on Audio, Speech and Language Processing*, 15(7):2072–2084, 2007.

[18] N. Brümmer, S. Cumani, O.Glembek, M. Karafiát, P. Matejka, J. Pesán, O. Plchot, M. Soufifar, E. de Villiers, and J. Cernocký. Description and analysis of the Brno 276 system for LRE2011. In *Odyssey 2012: The Speaker and Language Recognition Workshop*, pages 216–223, Singapore, June 2012.

[19] N. Brümmer and E. de Villiers. The BOSARIS Toolkit: Theory, Algorithms and Code for Surviving the New DCF. In *Proceedings of the NIST 2011 Speaker Recognition Workshop*, Atlanta (GA), USA, December 2011. http://sites.google.com/site/bosaristoolkit/.

[20] N. Brümmer and J. du Preez. Application-Independent Evaluation of Speaker Detection. *Computer, Speech and Language*, 20(2-3):230–275, April-July 2006.

[21] N. Brümmer, A. Strasheim, V. Hubeika, P. Matejka, L. Burget, and O. Glembek. Discriminative Acoustic Language Recognition via Channel-Compensated GMM Statistics. In *Proceedings of Interspeech*, pages 2187–2190, Brighton, UK, September 2009.

[22] N. Brümmer and D. van Leeuwen. On calibration of language recognition scores. In *Proceedings of Odyssey - The Speaker and Language Recognition Workshop*, pages 1–8, 2006.

[23] N. Brummer et. al. ABC System description for NIST SRE 2010. In *2010 NIST Speaker Recognition Evaluation (SRE)*, Brno, Czech Republic, 2010.

[24] L. Burget, O. Plchot, S. Cumani, O. Glembek, P. Matejka, and N. Brummer. Discriminatively trained probabilistic linear discriminant analysis for speaker verification. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2011*, pages 4832–4835. IEEE, 2011.

[25] W. Campbell, J. Campbell, T. Gleason, D. Reynolds, and W. Shen. Speaker verification using support vector machines and high-level features. *Audio, Speech, and Language Processing, IEEE Transactions on*, 15(7):2085–2094, September 2007.

[26] W. Campbell, T. Gleason, J. Navratil, D. Reynolds, W. Shen, E. Singer, and P. Torres-Carrasquillo. Advanced Language Recognition using Cepstra and Phonotactics: MITLL System Performance on the NIST 2005 Language Recognition Evaluation. In *Proceedings of Odyssey 2006 - The Speaker and Language Recognition Workshop*, pages 1–8, San Juan, Puerto Rico, June 2006.

[27] W. Campbell, D. Sturim, D. Reynolds, and A. Solomonoff. SVM Based Speaker Verification using a GMM Supervector Kernel and NAP Variability Compensation. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 1, pages 97–100, May 2006.

[28] W. M. Campbell. Generalized Linear Discriminant Sequence Kernels for Speaker Recognition. In *Proceedings of IEEE ICASSP*, volume I, pages 161–164, 2002.

[29] W. M. Campbell, F. Richardson, and D. A. Reynolds. Language Recognition with Word Lattices and Support Vector Machines. In *Proceedings of IEEE ICASSP*, pages 15–20, Honolulu, Hawaii, USA, 2007.

[30] W. M. Campbell, E. Singer, P. A. Torres-Carrasquillo, and D. A. Reynolds. Language Recognition with Support Vector Machines. In *Proceedings of Odyssey 2004 - The Speaker and Language Recognition Workshop*, pages 41–44, Toledo, Spain, May-June 2004.

[31] F. Castaldo, D. Colibro, S. Cumani, E. Dalmasso, P. Laface, and C. Vair. Loquendo-Politecnico di Torino system for the 2009 NIST Language Recognition Evaluation. In *ICASSP*, pages 5002–5005, 2010.

[32] F. Castaldo, D. Colibro, E. Dalmasso, P. Laface, and C. Vair. Compensation of Nuisance Factors for Speaker and Language Recognition. *IEEE Transactions on Audio, Speech and Language Processing*, 15(7):1969–1978, September 2007.

[33] F. Castaldo, D. Colibro, E. Dalmasso, P. Laface, and C. Vair. Acoustic Language Identification Using Fast Discriminative Training. In *Proceedings of Interspeech*, pages 346–349, Antwerp, Belgium, August 2007.

[34] F. Castaldo, S. Cumani, P. Laface, and D. Colibro. Language Recognition Using Language Factors. In *Proceedings of Interspeech*, pages 176–179, Brighton, UK, September 2009.

[35] J.-H. Chang, N. S. Kim, and S. K. Mitra. Voice activity detection based on multiple statistical models. *Signal Processing, IEEE Transactions on*, 54(6):1965–1976, June 2006.

[36] R. Collobert and S. Bengio. SVMTorch: Support vector machines for large-scale regression problems. *Journal of Machine Learning Research*, 1:143–160, 2001.

[37] H. Combrink and E. Botha. Automatic Language Identification: Performance vs Complexity. In *Proceedings of the Sixth Annual South Africa Workshop on Pattern Recognition*, 1997.

[38] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 28(4):357–366, Aug 1980.

[39] N. Dehak, P. Dumouchel, and P. Kenny. Modeling prosodic features with joint factor analysis for speaker verification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 15(7):2095–2103, 2007.

[40] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet. Front-end factor analysis for speaker verification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 19(4):788 –798, may 2011.

[41] N. Dehak, P. A. Torres-Carrasquillo, D. A. Reynolds, and R. Dehak. Language Recognition via i-vectors and Dimensionality Reduction. In *Proceedings of the Interspeech 2011*, pages 857–860, Florence, Italy, August 27-31 2011.

[42] A. P. Dempster, N. M. Laird, D. B. Rubin, et al. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal statistical Society*, 39(1):1–38, 1977.

[43] L. D'Haro, O.Glembek, O. Plchot, P. Matejka, M. Soufifar, R. Cordoba, and J. Cernocký. Phonotactic Language Recognition using i-vectors and Phoneme Posteriogram Counts. In *Proceedings of the Interspeech 2012*, Portland, Oregon, September 9-13 2012.

[44] M. Diez. *Verificación de la Lengua Mediante Modelos Acústicos*, June 2009. Final Year Project.

[45] M. Diez. *Compensación de canal en procesamiento del habla*, June 2010. MSc Thesis.

[46] M. Diez, M. Penagarikano, L. J. Rodriguez Fuentes, A. Varona, and G. Bordel. University of the Basque Country System for the 2011 NIST SRE Analysis Workshop. In *NIST 2011 Speaker Recognition Analysis Workshop*, Atlanta (USA), 8-9 december 2011.

[47] M. Diez, M. Penagarikano, A. Varona, L. J. Rodriguez Fuentes, and G. Bordel. GTTS System for the Albayzin 2010 Speaker Diarization Evaluation. In *VI Jornadas en Tecnologías del Habla and II Iberian SLTech Workshop*, pages 397–400, Vigo, Spain, 10-12 November 2010.

[48] M. Diez, M. Penagarikano, A. Varona, L. J. Rodriguez Fuentes, and G. Bordel. On the use of Dot Scoring for Speaker Diarization. In *Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2011)*, pages 612–619, Las Palmas de Gran Canaria. Spain., 8-10 June 2011.

[49] M. Diez, M. Penagarikano, A. Varona, L. J. Rodríguez-Fuentes, and G. Bordel. University of the Basque Country Systems for the NIST 2012 Speaker Recognition Evaluation. In *Proceedings of the NIST-SRE 2012*, Orlando, USA, December 11-12 2012.

[50] M. Diez, A. Varona, M. Penagarikano, L. J. Rodriguez-Fuentes, and G. Bordel. On the Use of Log-Likelihood Ratios as Features in Spoken Language Recognition. In *IEEE Workshop on Spoken Language Technology (SLT 2012)*, pages 274–279, Miami, Florida, USA, December 2012.

[51] M. Diez, A. Varona, M. Penagarikano, L. J. Rodríguez-Fuentes, and G. Bordel. Dimensionality Reduction of Phone Log-Likelihood Ratio Features for Spoken Language Recognition. In *Proceedings of Interspeech 2013*, pages 64–68, Lyon, France, August 2013.

[52] M. Diez, A. Varona, M. Penagarikano, L. J. Rodriguez Fuentes, and G. Bordel. Language Recognition on Albayzin 2010 LRE using PLLR features. *Procesamiento del Lenguaje Natural*, (51):153–160, 2013.

[53] M. Diez, A. Varona, M. Penagarikano, L. J. Rodríguez-Fuentes, and G. Bordel. Using Phone Log-Likelihood Ratios as Features for Speaker Recognition. In *Proceedings of Interspeech 2013*, Lyon, France, August 2013.

[54] M. Diez, A. Varona, M. Penagarikano, L. J. Rodriguez Fuentes, and G. Bordel. New Insight into the Use of Phone Log-Likelihood Ratios as Features for Language Recognition. In *Interspeech 2014*, Singapore, 14-18 sep. 2014.

[55] M. Diez, A. Varona, M. Penagarikano, L. J. Rodriguez Fuentes, and G. Bordel. On the Complementarity of Phone Posterior Probabilities for Improved Speaker Recognition. *IEEE Signal Processing Letters*, 21(6):649–652, jun. 2014.

[56] M. Diez, A. Varona, M. Penagarikano, L. J. Rodriguez Fuentes, and G. Bordel. On the Projection of PLLRs for Unbounded Feature Distributions in Spoken Language Recognition. *IEEE Signal Processing Letters*, 21(9):1073–1077, sep 2014.

[57] M. Diez, A. Varona, M. Penagarikano, L. J. Rodriguez Fuentes, and G. Bordel. Optimizing PLLR Features for Spoken Language Recognition. In *proceedings of the 22nd International Conference on Pattern Recognition (ICPR'14)*, pages 779–784, Stockholm, Sweden, 24-28 aug 2014.

[58] G. R. Doddington. Speaker recognition based on idiolectal differences between speakers. In *Proceedings of Interspeech 2001*, pages 2521–2524, 2001.

[59] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008. Software available at http://www.csie.ntu.edu.tw/~cjlin/liblinear.

[60] L. Ferrer, M. McLaren, N. Scheffer, Y. Lei, M. Graciarena, and V. Mitra. A noise-robust system for nist 2012 speaker recognition evaluation. In *INTERSPEECH*, pages 1981–1985, 2013.

[61] FoCal. *Tools for evaluation, calibration and fusion of, and decision-making with, multi-class statistical pattern recognition scores*, 2007. https://sites.google.com/site/nikobrummer/focalmulticlass.

[62] FoCal. *Toolkit for Evaluation, Fusion and Calibration of statistical pattern recognizers*, 2008. http://sites.google.com/site/nikobrummer/focal.

[63] S. Ganapathy, S. Thomas, and H. Hermansky. Feature Extraction Using 2-D Autoregressive Models For Speaker Recognition. In *ISCA Speaker Odyssey*, 2012.

[64] D. Garcia-Romero and C. Y. Epsy-Wilson. Analysis of I-vector Length Normalization in Speaker Recognition Systems. In *INTERSPEECH*, pages 249–252, Florence, Italy, August 2011.

[65] J. L. Gauvain and C. Lee. Maximum A Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains. *IEEE Transactions on Speech and Audio Processing*, 2:291–298, 1994.

[66] J. L. Gauvain, A. Messaoudi, and H. Schwenk. Language recognition using phone lattices. In *Proceedings of ICSLP*, pages 1283–1286, 2004.

[67] P. Ghosh, A. Tsiartas, and S. Narayanan. Robust voice activity detection using long-term signal variability. *Audio, Speech, and Language Processing, IEEE Transactions on*, 19(3):600–613, March 2011.

[68] L. Gillick and S. Cox. Some statistical issues in the comparison of speech recognition algorithms. In *Proc. ICASSP*, pages 532–535, 1989.

[69] C. Greenberg, A. Martin, and M. Przybocki. The 2011 NIST Language Recognition Evaluation. In *Proceedings of Interspeech*, Portland, Oregon, 2012.

[70] C. S. Greenberg, V. M. Stanford, A. F. Martin, M. Yadagiri, G. R. Doddington, J. J. Godfrey, and J. Hernandez-Cordero. The 2012 NIST Speaker Recognition Evaluation. In *INTERSPEECH*, pages 1971–1975, 2013.

[71] F. J. Harris. On the use of windows for harmonic analysis with the discrete Fourier transform. *Proc. of the IEEE*, 66(1):51–83, 1978.

[72] T. Hasan, S. Sadjadi, G. Liu, N. Shokouhi, H. Boril, and J. Hansen. Crss systems for 2012 nist speaker recognition evaluation. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 6783–6787, May 2013.

[73] H. Hermansky. Perceptual linear predictive (PLP) analysis of speech. *J. Acoust. Soc. Am.*, 57(4):1738–52, Apr. 1990.

[74] H. Hermansky and N. Morgan. RASTA Processing of Speech. *Speech and Audio Processing, IEEE Transactions on*, 2(4):578 –589, October 1994.

[75] V. Hubeika, L. Burget, P. Matejka, and P. Schwarz. Discriminative training and channel compensation for acoustic language recognition. In *Proc. Interspeech 2008*, 2008.

[76] S. Itahashi and L. Du. Language identification based on speech fundamental frequency. In *Proceedings of IEEE ICASSP*, volume 2, pages 1359–1362, September 1995.

[77] Z. Jancík, O. Plchot, N. Brummer, L. Burget, O. Glembek, V. Hubeika, M. Karafiát, P. Matejka, T. Mikolov, A. Strasheim, and J. Cernocký. Data selection and calibration issues in automatic language recognition - investigation with BUT-AGNITIO NIST LRE 2009 system. In *Proc. Odyssey 2010 - The Speaker and Language Recognition Workshop*, pages 215–221, 2010.

[78] T. Kamm, H. Hermansky, and A. G. Andreou. *Learning the Melscale and Optimal VTN Mapping*, 1997. Technical report. CSLP.

[79] T. Kempton and R. K. Moore. Language Identification: Insights from the Classification of Hand Annotated Phone Transcripts. In *Proc. Odyssey 2008 - The Speaker and Language Recognition Workshop*, Stellenbosch, South Africa, January 21-24 2008.

[80] P. Kenny. Joint factor analysis of speaker and session variability: Theory and algorithms. Technical Report Technical Report CRIM-06/08-13, CRIM, 2005. [Online]. Available: http://www.crim.ca/perso/patrick.kenny/.

[81] P. Kenny. Bayesian speaker verification with heavy-tailed priors. In *Odyssey, The Speaker and Language Tecognition Workshop*, 2010.

[82] P. Kenny and P. Dumouchel. Experiments in speaker verification using factor analysis likelihood ratios. In *in Odyssey: The Speaker and Language Recognition Workshop*, pages 219–226, 2004.

[83] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel. A study of interspeaker variability in speaker verification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 16(5):980–988, July 2008.

[84] E. Khoury, B. Vesnicer, J. Franco-Pedroso, R. Violato, Z. Boulkenafet, L. M. Fernandez, M. Diez, J. Kosmala, H. Khemiri, T. Cipr, M. G. R. Saeidi, J. Zganec-Gros, R. Z. Candil, F. Simoes, M. Bengherabi, A. A. Marquina, M. Penagarikano, A. Abad, M. Boulayemen, P. Schwarz, D. V. Leeuwen, J. Gonzalez-Domínguez, M. U. Neto, E. Boutellaa, P. G. Vilda, A. Varona, D. Petrovska-Delacretaz, P. Matejka, J. Gonzalez-Rodríguez, T. Pereira, F. Harizi, L. J. Rodriguez Fuentes, L. E. Shafey, M. Angeloni, G. Bordel, G. Chollet, and S. Marcel. The 2013 speaker recognition evaluation in mobile environment. In *The 6th IAPR International Conference on Biometrics (ICB-2013)*, pages 1–8, Madrid, Spain., June 4 - 7 2013.

[85] T. Kinnunen and R. González-Hautamäki. Long-term f0 modeling for text-independent speaker recognition. In *Proceedings of the 10th International Conference Speech and Computer (SPECOM), Patras, Greece*, pages 567–570, 2005.

[86] T. Kinnunen and H. Li. An overview of text-independent speaker recognition: From features to supervectors. *Speech Communication*, 52(1):12 – 40, 2010.

[87] M. Kockmann, L. Ferrer, L. Burget, and J. Cernocký. ivector fusion of prosodic and cepstral features for speaker verification. In *INTERSPEECH*, pages 265–268, 2011.

[88] C. S. Kumar, H. Li, R. Tong, P. Matejka, L. Burget, and J. Cernocký. Tuning Phone Decoders For Language Identification. In *Proceedings of the workshop*

on Human Language Technology, pages 4861–4864, Stroudsburg, PA, USA, 2010.

[89] F.-H. Liu, R. M. Stern, X. Huang, and A. Acero. Efficient cepstral normalization for robust speech recognition. In *Proceedings of the workshop on Human Language Technology*, HLT '93, pages 69–74, Stroudsburg, PA, USA, 1993. Association for Computational Linguistics.

[90] S. H. Mallidi, S. Ganapathy, and H. Hermansky. Robust Speaker Recognition Using Spectro-Temporal Autoregressive Models. In *Proceedings of Interspeech 2013*, number 8, pages 3689–3693. International Speech Communication Association, 2013.

[91] S. Marcel, C. McCool, P. Matejka, T. Ahonen, J. Cernocky, and al. On the results of the first mobile biometry (mobio) face and speaker verification evaluation. Idiap-RR Idiap-RR-30-2010, 8 2010.

[92] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki. The DET Curve in Assessment of Detection Task Performance. In *Proceedings of Eurospeech*, pages 1985–1988, 1997.

[93] A. Martin and C. Greenberg. The 2009 NIST Language Recognition Evaluation. In *Odyssey 2010 - The Speaker and Language Recognition Workshop, paper 030*, pages 165–171, Brno, Czech Republic, 2010.

[94] A. F. Martin and C. S. Greenberg. The NIST 2010 Speaker Recognition Evaluation. In *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[95] A. F. Martin and A. N. Le. The Current State of Language Recognition. In *Proceedings of Odyssey 2006 - The Speaker and Language Recognition Workshop*, 2006.

[96] A. F. Martin and A. N. Le. NIST 2007 Language Recognition Evaluation. In *Proceedings of Odyssey 2008 - The Speaker and Language Recognition Workshop*, 2008.

[97] D. Martínez, L. Burget, L. Ferrer, and N. Scheffer. iVector-based Prosodic System for Language Identification. In *Proceedings of ICASSP*, pages 4861–4864, Japan, 2012.

[98] D. Martínez, O. Plchot, L. Burget, O. Glembek, and P. Matejka. Language Recognition in iVectors Space. In *Proceedings of Interspeech*, pages 861–864, Firenze, Italy, 2011.

[99] P. Matejka, O. Plchot, M. Soufifar, O. Glembek, L. F. D'Haro, K. Veselý, F. Grézl, J. Z. Ma, S. Matsoukas, and N. Dehak. Patrol Team Language Identification System for DARPA RATS P1 Evaluation. In *Proceedings of the Interspeech 2012*, Portland, Oregon, September 9-13 2012.

[100] P. Matějka, O. Glembek, F. Castaldo, M. Alam, O. Plchot, P. Kenny, L. Burget, and J. Černocký. Full-covariance UBM and heavy-tailed PLDA in i-vector speaker verification. In *Proc. ICASSP*, 2011.

[101] C. McCool, S. Marcel, A. Hadid, M. Pietikainen, P. Matejka, J. Cernocky, N. Poh, J. Kittler, A. Larcher, C. Levy, D. Matrouf, J.-F. Bonastre, P. Tresadern, and T. Cootes. Bi-modal person recognition on a mobile phone: using mobile phone data. In *IEEE ICME Workshop on Hot Topics in Mobile Multimedia*, July 2012.

[102] H. Melin. Databases For Speaker Recognition: Activities In COST250 Working Group 2. In *in Proceedings COST250 Workshop on Speaker Recognition in Telephony*, 1999.

[103] R. W. M. Ng, T. Lee, C.-C. Leung, B. Ma, and H. Li. Spoken Language Recognition With Prosodic Features. *IEEE Transactions on Audio, Speech & Language Processing*, 21(9):1841–1853, Sept 2013.

[104] J. Pelecanos and S. Sridharan. Feature Warping for Robust Speaker Verification. In *2001: A Speaker Odyssey - The Speaker Recognition Workshop*, pages 213–218, 2001.

[105] M. Penagarikano and G. Bordel. Sautrela: a highly modular open source speech recognition framework. In *Automatic Speech Recognition and Understanding, 2005 IEEE Workshop on*, pages 386–391, 2005. http://gtts.ehu.es/TWiki/bin/view/Sautrela.

[106] M. Penagarikano, A. Varona, M. Diez, L. J. Rodriguez Fuentes, and G. Bordel. A speaker recognition system based on sufficient-statistics-space channel-compensation and dot-scoring. In *VI Jornadas en Tecnologías del Habla and II Iberian SLTech Workshop*, pages 135–138, Vigo, Spain, 10-12 November 2010.

[107] M. Penagarikano, A. Varona, M. Diez, L. J. Rodriguez Fuentes, and G. Bordel. University of the Basque Country System for NIST 2010 Speaker Recognition Evaluation. In *2010 NIST Speaker Recognition Evaluation (SRE) Workshop*, Brno, Czech Republic, 24-25 June 2010.

[108] M. Penagarikano, A. Varona, M. Diez, L. J. Rodriguez Fuentes, and G. Bordel. University of the Basque Country System for NIST 2010 Speaker Recognition Evaluation. In *V Jornadas de Reconocimiento Biométrico de Personas*, Huesca, Spain, 2-3 September 2010.

[109] M. Penagarikano, A. Varona, M. Diez, L. J. Rodriguez Fuentes, and G. Bordel. Study of Different Backends in a State-Of-the-Art Language Recognition System. In *Interspeech 2012*, Portland, Oregon, USA, 9-13 September 2012.

[110] M. Penagarikano, A. Varona, L. Rodríguez-Fuentes, and G. Bordel. A dynamic approach to the selection of high-order n-grams in phonotactic language recognition. In *Proceedings of ICASSP*, pages 4412–4415, Prague, Czech Republic, May 22-27 2011.

[111] M. Penagarikano, A. Varona, L. Rodriguez-Fuentes, and G. Bordel. Dimensionality Reduction for Using High-Order n-grams in SVM-Based Phonotactic Language Recognition. In *Proceedings of Interspeech 2011*, pages 853–856, Florence, Italy, August 28-31 2011.

[112] M. Penagarikano, A. Varona, L. J. Rodríguez-Fuentes, and G. Bordel. Improved Modeling of Cross-Decoder Phone Co-occurrences in SVM-based Phonotactic Language Recognition. *IEEE Transactions on Audio, Speech and Language Processing*, 19(8):2348–2363, November 2011.

[113] M. Penagarikano, A. Varona, L. J. Rodriguez-Fuentes, M. Diez, and G. Bordel. University of the Basque Country (EHU) Systems for the 2011 NIST Language Recognition Evaluation. In *Proceedings of the NIST 2011 LRE Workshop*, Atlanta (USA), 6-7 december 2011.

[114] M. Penagarikano, A. Varona, L. J. Rodriguez Fuentes, M. Diez, and G. Bordel. The EHU Systems for the NIST 2011 Language Recognition Evaluation. In *Interspeech 2012*, Portland, Oregon, USA, 9-13 September 2012.

[115] D. Percival and A. Walden. *Spectral Analysis for Physical Applications*. Cambridge University Press, Cambridge, UK, 1993.

[116] O. Plchot, M. Diez, and M. S. L. Burget. PLLR Features in Language Recognition System for RATS. In *Proceedings of Interspeech 2014*, Singapore, September 2014.

[117] O. Plchot, M. Karafiát, N. Brümmer, O. Glembek, P. Matejka, and E. de Villiers J. Cernocký. Speaker vectors from Subspace Gaussian Mixture Model as complementary features for Language Identification. In *Odyssey 2012: The Speaker and Language Recognition Workshop*, pages 330–333, Singapore, June 2012.

[118] S. J. D. Prince and J. H. Elder. Probabilistic Linear Discriminant Analysis for Inferences About Identity. In *ICCV*, pages 1–8, 2007.

[119] D. Reynolds, W. Andrews, J. Campbell, J. Navratil, B. Peskin, A. Adami, Q. Jin, D. Klusacek, J. Abramson, R. Mihaescu, J. Godfrey, D. Jones, and B. Xiang. The supersid project: exploiting high-level information for high-accuracy speaker recognition. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on*, volume 4, pages IV–784–7 vol.4, April.

[120] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker Verification Using Adapted Gaussian Mixture Models. *Digital Signal Processing*, 10(1-3):19–41, January 2000.

[121] F. Richardson and W. Campbell. Language recognition with discriminative keyword selection. In *Proceedings of ICASSP*, pages 4145–4148, 2008.

[122] F. S. Richardson and W. M. Campbell. NAP for high level language identification. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2011)*, pages 4392–4395, Prague, Czech Republic, May 22-27 2011.

[123] L. J. Rodriguez Fuentes, N. Brümmer, M. Penagarikano, A. Varona, G. Bordel, and M. Diez. The Albayzin 2012 Language Recognition Evaluation. In *Interspeech 2013*, Lyon, France, 25-29 aug. 2013.

[124] L. J. Rodriguez Fuentes, N. Brümmer, M. Penagarikano, A. Varona, M. Diez, and G. Bordel. The Albayzin 2012 Language Recognition Evaluation Plan. In *iberspeech 2012*, madrid, spain., 21-23 nov. 2012.

[125] L. J. Rodriguez-Fuentes, M. Penagarikano, G. Bordel, and A. Varona. The Albayzin 2008 Language Recognition Evaluation. In *Proceedings of Odyssey 2010: The Speaker and Language Recognition Workshop*, pages 172–179, Brno, Czech Republic, 28 June - 1 July 2010.

[126] L. J. Rodriguez-Fuentes, M. Penagarikano, G. Bordel, A. Varona, and M. Diez. KALAKA: A TV Broadcast Speech Database for the Evaluation of Language Recognition Systems. In *Proceedings of the LREC 2010*, pages 1678–1685, Valleta, Malta, 17-23 May 2010.

[127] L. J. Rodriguez Fuentes, M. Penagarikano, A. Varona, M. Diez, and G. Bordel. Overview of the Albayzin 2010 Language Recognition Evaluation: database design, evaluation plan and preliminary analysis of results. In *VI Jornadas en Tecnologías del Habla and II Iberian SLTech Workshop*, pages 309–316, Vigo, Spain, 10-12 November 2010.

[128] L. J. Rodriguez-Fuentes, M. Penagarikano, A. Varona, M. Diez, and G. Bordel. The Albayzin 2010 Language Recognition Evaluation. In *Proceedings of Interspeech*, pages 1529–1532, Firenze, Italia, August 28-31 2011.

[129] L. J. Rodriguez-Fuentes, M. Penagarikano, A. Varona, M. Diez, and G. Bordel. KALAKA-2: a TV broadcast speech database for the recognition of Iberian languages in clean and noisy environments. In *Proceedings of the LREC*, Istanbul, Turkey, 23-25 May 2012.

[130] L. J. Rodriguez Fuentes, M. Penagarikano, A. Varona, M. Diez, and G. Bordel. KALAKA-3: a database for the recognition of spoken European languages on YouTube audios. In *Proceedings of the 9th international conference on language resources and evaluation (LREC'14)*, Reykjavik, Iceland, 26-31 may 2014. European Language Resources Association (ELRA).

[131] L. J. Rodriguez Fuentes, M. Penagarikano, A. Varona, M. Diez, G. Bordel, A. Abad, D. Martinez, J. Villalba, A. Ortega, and E. Lleida. The BLZ Systems for the 2011 NIST Language Recognition Evaluation. In *NIST 2011 Language Recognition Evaluation Workshop*, Atlanta (USA), 6-7 december 2011.

[132] L. J. Rodriguez Fuentes, M. Penagarikano, A. Varona, M. Diez, G. Bordel, A. Abad, D. Martinez, J. Villalba, A. Ortega, and E. Lleida. The BLZ Submission to the NIST 2011 LRE: Data Collection, System Development and Performance. In *Interspeech 2012*, Portland, Oregon, USA, 9-13 September 2012.

[133] L. J. Rodriguez-Fuentes, M. Penagarikano, A. Varona, M. Diez, G. Bordel, D. Martínez, J. Villalba, A. Miguel, A. Ortega, E. Lleida, A. Abad, O. Koller, I. Trancoso, P. Lopez-Otero, L. Docio-Fernandez, C. Garcia-Mateo, R. Saeidi, M. Soufifar, T. Kinnunen, T. Svendsen, and P. Franti. Multi-site Heterogeneous System Fusions for the Albayzin 2010 Language Recognition Evaluation. In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU) 2011*, Hawaii, USA, December 2011.

[134] L. J. Rodriguez Fuentes, A. Varona, M. Diez, M. Penagarikano, and G. Bordel. Evaluation of Spoken Language Recognition Technology Using Broadcast Speech: Performance and Challenges. In *Odyssey 2012: The Speaker and Language Recognition Workshop*, Singapore, June 25-28, 2012.

[135] L. J. Rodriguez Fuentes, A. Varona, M. Penagarikano, M. Diez, and G. Bordel. Spoken language recognition in conversational telephone speech and TV broadcast news (GLOSA). In *XXVI Congreso de la Sociedad Española para el Procesamiento de Lenguaje Natural (SEPLN)*, Huelva, Spain, 5-7 September 2011.

[136] A. Rosenberg, C. Lee, and F. Soong. Cepstral Channel Normalization Techniques for HMM-based Speaker Verification. In *Proceedings of ICSPL*, pages 1835–1838, 1994.

[137] P. Schwarz. *Phoneme recognition based on long temporal context*. PhD thesis, Faculty of Information Technology, Brno University of Technology, http://www.fit.vutbr.cz/, Brno, Czech Republic, 2008.

[138] E. Shriberg, L. Ferrer, S. Kajarekar, A. Venkataraman, and A. Stolcke. Modeling prosodic feature sequences for speaker recognition. *Speech Communication*, 46(3):455–472, 2005.

[139] E. Singer, P. A. Torres-Carrasquillo, D. A. Reynolds, A. McCree, F. Richardson, N. Dejak, and D.Sturim. The MITLL NIST LRE 2011 Language Recognition System. In *Odyssey 2012: The Speaker and Language Recognition Workshop*, pages 209–215, Singapore, June 2012.

[140] J. Sohn, N. S. Kim, and W. Sung. A statistical model-based voice activity detection. *Signal Processing Letters, IEEE*, 6(1):1–3, Jan 1999.

[141] A. Solomonoff, W. Campbell, and I. Boardman. Advances In Channel Compensation For SVM Speaker Recognition. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, volume 1, pages 629–632, March 2005.

[142] M. Soufifar, L. Burget, O. Plchot, S. Cumani, and J. Cernocky. Regularized subspace n-gram model for phonotactic ivector extraction. In *Proceedings of Interspeech*, Lyon, France, 2013.

[143] M. Soufifar, S. Cumani, L. Burget, and J. H. Cernocky. Discriminative Classifiers for Phonotactic Language Recognition with iVectors. In *Proceedings of ICASSP*, pages 4853–4856, Kyoto, Japan, 2012.

[144] A. Stolcke. SRILM - An extensible language modeling toolkit. In *Proceedings of Interspeech*, pages 257–286, November 2002.

[145] A. Strasheim and N. Brümmer. SUNSDV system description: NIST SRE 2008. In *NIST 2008 Speaker Recognition Evaluation Workshop Booklet*, 2008.

[146] R. Tong, B. Ma, H. Li, and E. S. Chng. A Target-Oriented Phonotactic Front-End for Spoken Language Recognition. *IEEE Transactions on Audio, Speech and Language Processing*, 17(7):1335–1347, September 2009.

[147] R. Tong, B. Ma, H. Li, and E. S. Chng. Selecting Phonotactic Features for Language Recognition. In *Proceedings of Interspeech*, pages 737–740, September 2010.

[148] P. Torres-Carrasquillo, E. Singer, W. Campbell, T. Gleason, A. McCree, D. Reynolds, F. Richardson, W. Shen, and D. Sturim. The MITLL NIST LRE 2007 language recognition system. In *Proceedings of Interspeech*, pages 719–722, 2008.

[149] P. A. Torres-Carrasquillo, E. Singer, M. A. Kohler, R. J. Greene, D. A. Reynolds, and J. R. Deller. Approaches to Language Identification using Gaussian Mixture Models and Shifted Delta Cepstral Features. In *Proceedings of ICSLP*, pages 89–92, 2002.

[150] D. A. Van Leeuwen and N. Brummer. Channel-dependent gmm and multi-class logistic regression models for language recognition. In *Speaker and Language Recognition Workshop, 2006. IEEE Odyssey 2006: The*, pages 1–8. IEEE, 2006.

[151] A. Vandecatseye, J.-P. Martens, J. P. Neto, H. Meinedo, C. Garcia-Mateo, J. Dieguez-Tirado, F. Mihelic, J. Zibert, J. Nouza, P. David, M. Pleva, A. Cizmar, H. Papageorgiou, and C. Alexandris. The COST278 pan-european broadcast news database. In *Proceedings of LREC*, Lisbon, Portugal, 2004.

[152] A. Varona, S. Nieto, L. J. Rodriguez Fuentes, M. Penagarikano, G. Bordel, and M. Diez. A Spoken Document Retrieval System for TV Broadcast News in Spanish and Basque. In *XXVI Congreso de la Sociedad Española para el Procesamiento de Lenguaje Natural (SEPLN)*, Huelva, Spain, 5-7 September 2011.

[153] J. A. Villalba and E. Lleida. Handling i-vectors from different recording conditions using multi-channel simplified PLDA in speaker recognition. In *ICASSP*, pages 6763–6767, 2013.

[154] H. Wang, C.-C. Leung, T. Lee, B. Ma, and H. Li. Shifted-Delta MLP Features for Spoken Language Recognition. *IEEE Signal Process. Lett.*, 20(1):15–18, 2013.

[155] E. Wong, J. Pelecanos, S. Myers, and S. Sridharan. Language Identification Using Efficient Gaussian Mixture Model Analysis. In *Proceedings of the Australian International Conference on Speech Scienc and Tehcnology*, 2000.

[156] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Lui, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. *The HTK Book (for HTK Version 3.4)*. Entropic, Ltd., Cambridge, UK, 2006.

[157] A. Zgank, B. Horvat, and Z. Kacic. Data-driven generation of phonetic broad classes, based on phoneme confusion matrix similarity. *Speech Comunication*, 47(3):379–393, September 2005.

[158] Q. Zhang, G. Liu, and J. H. Hansen. Robust Language Recogntion Based on Diverse Features. In *Odyssey 2014: The Speaker and Language Recognition Workshop*, pages 152–157, Joensuu, Finland, June 2014.

[159] M. Zissman. Comparison of Four Approaches to Automatic Language Identification of Telephone Speech. *IEEE Transactions on Speech and Audio Processing*, 4(1):31–44, January 1996.

[160] M. A. Zissman and K. M. Berkling. Automatic language identification. *Speech Communication*, 35(1):115–124, 2001.

[161] V. Zue, S. Seneff, and J. R. Glass. Speech database development at MIT: TIMIT and beyond. *Speech Communication*, 9(4):351–356, 1990.