TESIS DE LA UNIVERSIDAD DE ZARAGOZA
2021
77

Ignacio Viñals Bailo

Advances in Subspace-based Solutions for Diarization in the Broadcast Domain

Director/es
Ortega Giménez, Alfonso

ISSN 2254-7606

Prensas de la Universidad
Universidad Zaragoza

ISSN 2254-7606

Universidad Zaragoza
1542

Tesis Doctoral

ADVANCES IN SUBSPACE-BASED SOLUTIONS FOR DIARIZATION IN THE BROADCAST DOMAIN

Autor

Ignacio Viñals Bailo

Director/es

Ortega Giménez, Alfonso

UNIVERSIDAD DE ZARAGOZA
Escuela de Doctorado

Programa de Doctorado en Tecnologías de la Información y Comunicaciones en Redes Móviles

2020

UNIVERSIDAD DE ZARAGOZA
TESIS DOCTORAL - INGENIERÍA DE TELECOMUNICACIÓN

Advances in Subspace-based Solutions for Diarization in the Broadcast Domain

Author:
Ignacio Viñals Bailo

Supervisor:
Alfonso Ortega Giménez

DEPARTAMENTO DE INGENIERÍA ELECTRÓNICA Y COMUNICACIONES
ESCUELA DE INGENIERÍA Y ARQUITECTURA
April, 2020

Universidad Zaragoza

A mis padres

The human voice is
the most perfect instrument of all.

Arvo Pärt

The most important questions of life are,
for the most part,
really only problems of probability

Pierre-Simon Laplace

Research is creating new knowledge.

Neil Armstrong

The human brain is an incredible
pattern-matching machine.

Jeff Bezos

Acknowledgements

It has been five long years since I made the decision to start a PhD programme. Along all this time I have been fortunate enough to meet, collaborate and be helped as well as supported by many people, without whom this thesis would not be available today. These lines are dedicated to all of them.

First and foremost, I want to dedicate some lines to my parents. I want to thank them for being on my side from the very beginning. They always offered me their support since I decided to become a researcher. For the last five years they have cheered me up during bad times and kept my feet on the ground during those limited successful occasions. This work could not be possible without their contribution.

Besides, I also must thank Alfonso Ortega for giving me the chance to grow up professionally and personally. He gave me the chance to discover the researcher career when I was undergraduate, and offered me the opportunity to keep on developing myself with ViVoLAB group. For the last five years he has become a friend apart from a supervisor, who guided me along this difficult learning process. By means of our meetings he helped me to discover some of the best ideas while other times he simply made me aware that sometimes I could not see the forest for the trees.

Apart from Alfonso Ortega, ViVoLAB group is also full of wonderful people who deserve their own mention. Some of my kindest memories along my PhD years include Eduardo Lleida. He always tried to build a group based on friendship relationships rather than simply professional ones. Besides, it is praiseworthy how this treatment also included ViVoLAB alumni all over the world. Another important collaborator in this thesis is Antonio Miguel. His valuable expertise in subspace models and neural networks were capital for the work done in this thesis. Besides, the theoretical discussions we had about these topics were also very enriching, opening my eyes about unseen lines of research to explore.

I also want to acknowledge Johns Hopkins university staff, specially Najim Dehak and his

group, for the research visit I could do in 2017. This visit allowed me to work in new lines of research while participating in the SCALE workshop. This chance made possible to work with Jesús Villalba, about whom I also have some words. While our first time in touch was *original*, later on he demonstrated me his quality as researcher and specially friend. He became some sort of role model, introducing me into the Bayesian techniques, being a pleasure learning from him. In USA I also had the chance to closely know my dear Paola. Although we first met when I joined ViVoLAB group (I will never forget how she tried to send me abroad in my first week with ViVoLAB) and kept in touch for the following years, it was in Baltimore when I could closely share some time with her. I absolutely thank her for her natural kindness, receiving me every day with a smile and making my life more comfortable during my visit.

Apart from the fabulous people I could meet at work, I should not forget about my friends. Too many names come to my mind, so I prefer not listing their names in order not to forget anybody. They offered me their friendship, sharing together all these fun situations when I could simply disconnect. Simple plans such a film, having dinner or a coffee together were outstanding to gain energy in order to come back to the lab the day after.

Finally, my dear Laura also deserves her own mention. Since we met I have always felt that she perfectly complemented me in all manners, becoming my confidant and support and providing me the balance to deal with the difficulties, both personally and professionally. She encouraged me to always do my best with a smile in the face. Thanks Laura for everything.

Abstract

The motivation for this thesis is the need for robust diarization solutions. These diarization techniques must add value to the increasing amount of available multimedia data by accurately discriminating among the speakers present in the audio signal. Unfortunately, up to recent times this type of technology was only viable in restricted conditions, far from a general solution.

The reasons why diarization performance is limited are multiple. The first reason to take into account is the high complexity of the speech generation process, in particular the physiological procedures to include the discriminative speaker characteristics in the voice signal. This complexity makes the reverse process, i.e. the estimation of these characteristics from raw speech, an inefficient task by means of the current state of the art. Thus, approximations should be considered instead. The efforts in the modeling task have provided more and more elaborated solutions, despite not trending towards the explanation of the physiological causes. Rather than learning the biological rules of speech, these models learn acoustic relationships within a large data training pool. This development of approximated models generates the second reason, the domain variability. Due to the fact that we are exploiting local relationships learnt from a specific data training pool, when moving to a different domain with different conditions these learnt relationships may differ, causing systems to significantly fail.

Our contribution for diarization technologies has been focused on the broadcast domain. This domain is currently a challenging scenario for diarization systems where no limitations can be considered. Therefore, we should learn how to efficiently model the audio in order to extract the most useful information possible and how to obtain the speaker labels accordingly. Moreover, the presence of multiple audio conditions due to different shows and genres requires the development of techniques capable of adapting the knowledge acquired from a certain domain where data is available to those domains where these data is either scarce or simply unavailable.

For this purpose, the work developed in this thesis has focused on three main subtasks, speaker characterization, clustering and model adaptation. The first subtask seeks the modeling

of a certain piece of audio in order to obtain an accurate representation of the involved speakers, highlighting their discriminative properties. In this area a study about the current modeling strategies has been done, paying attention to the limitations of the obtained representations and exposing the type of errors they can generate. Besides, DNN alternatives making use of this knowledge were also proposed. This line of research is responsible for one JCR article as well as three papers presented in international conferences. The second step is clustering, in charge of strategies to search the optimal speaker label arrangement. The carried out research has proposed novel strategies to estimate the best partition of speakers based on subspace techniques, specially PLDA, generating two papers presented in international conferences. Finally, the model adaptation task seeks transferring the knowledge acquired from a data training pool to alternative domains from which no further knowledge is available. For this purpose, our work has focused on the extraction of inferred speaker information from the audio to diarize, which is later used for the adaptation of the involved models. This line of research is responsible for one JCR article as well as two papers in international conferences.

Resumen

La motivación de esta tesis es la necesidad de soluciones robustas al problema de diarización. Estas técnicas de diarización deben proporcionar valor añadido a la creciente cantidad disponible de datos multimedia mediante la precisa discriminación de los locutores presentes en la señal de audio. Desafortunadamente, hasta tiempos recientes este tipo de tecnologías solamente era viable en condiciones restringidas, quedando por tanto lejos de una solución general.

Las razones detrás de las limitadas prestaciones de los sistemas de diarización son múltiples. La primera causa a tener en cuenta es la alta complejidad de la producción de la voz humana, en particular acerca de los procesos fisiológicos necesarios para incluir las características discriminativas de locutor en la señal de voz. Esta complejidad hace del proceso inverso, la estimación de dichas características a partir del audio, una tarea ineficiente por medio de las técnicas actuales del estado del arte. Consecuentemente, en su lugar deberán tenerse en cuenta aproximaciones. Los esfuerzos en la tarea de modelado han proporcionado modelos cada vez más elaborados, aunque no buscando la explicación última de naturaleza fisiológica de la señal de voz. En su lugar estos modelos aprenden relaciones entre las señales acústicas a partir de un gran conjunto de datos de entrenamiento. El desarrollo de modelos aproximados genera a su vez una segunda razón, la variabilidad de dominio. Debido al uso de relaciones aprendidas a partir de un conjunto de entrenamiento concreto, cualquier cambio de dominio que modifique las condiciones acústicas con respecto a los datos de entrenamiento condiciona las relaciones asumidas, pudiendo causar fallos consistentes en los sistemas.

Nuestra contribución a las tecnologías de diarización se ha centrado en el entorno de radiodifusión. Este dominio es actualmente un entorno todavía complejo para los sistemas de diarización donde ninguna simplificación de la tarea puede ser tenida en cuenta. Por tanto, se deberá desarrollar un modelado eficiente del audio para extraer la información de locutor y como inferir el etiquetado correspondiente. Además, la presencia de múltiples condiciones acústicas debido a la existencia de diferentes programas y/o géneros en el dominio requiere

el desarrollo de técnicas capaces de adaptar el conocimiento adquirido en un determinado escenario donde la información está disponible a aquellos entornos donde dicha información es limitada o sencillamente no disponible.

Para este propósito el trabajo desarrollado a lo largo de la tesis se ha centrado en tres subtareas: caracterización de locutor, agrupamiento y adaptación de modelos. La primera subtarea busca el modelado de un fragmento de audio para obtener representaciones precisas de los locutores involucrados, poniendo de manifiesto sus propiedades discriminativas. En esta área se ha llevado a cabo un estudio acerca de las actuales estrategias de modelado, especialmente atendiendo a las limitaciones de las representaciones extraídas y poniendo de manifiesto el tipo de errores que pueden generar. Además, se han propuesto alternativas basadas en redes neuronales haciendo uso del conocimiento adquirido. Esta línea de investigación ha generado un artículo JCR y tres contribuciones en conferencias internacionales. La segunda tarea es el agrupamiento, encargado de desarrollar estrategias que busquen el etiquetado óptimo de los locutores. La investigación desarrollada durante esta tesis ha propuesto nuevas estrategias para estimar el mejor reparto de locutores basadas en técnicas de subespacios, especialmente PLDA, generando dos contribuciones en conferencias internacionales. Finalmente, la tarea de adaptación de modelos busca transferir el conocimiento obtenido de un conjunto de entrenamiento a dominios alternativos donde no hay datos para extraerlo. Para este propósito los esfuerzos se han centrado en la extracción no supervisada de información de locutor del propio audio a diarizar, siendo posteriormente usada en la adaptación de los modelos involucrados. Esta línea de investigación es responsable de un artículo JCR, así como de dos contribuciones en conferencias internacionales.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years Broadcast data has experienced a huge evolution on its business. The traditional broadcast contents, cinema, radio and television, only in the USA, involve more than 1700 TV channels[1], more than 31000 radio stations[2] and 800 films produced every year. In addition to these figures, we now must take into account the recently popular Video on Demand (VOD). This new business formula (Watch whatever you want whenever you want) was firstly originated to share videos (Youtube), but soon has become the newest business opportunity for the major content producers. Currently many well-known corporations are interested in this new technology, with active platforms (Netflix, Hulu, Amazon, etc) and some still in development. The impact of VoD in the Broadcast media is increasing year by year. Some of these services can provide up to 140 million hours of multimedia content per day [3]. Moving towards "amateur" production, figures are even higher. Platforms such as Youtube or Twitch must handle more than a billion hours of daily seen content [4] and more than 3.2 million broadcasters can be involved in this production.

Due to the large multimedia offer and the competitiveness among the content providers the need for original content has increased up to numbers never seen before. In order to properly manage this enormous amount of data, any new multimedia content should be accompanied by some extra information metadata. This information can consist of audio transcription, topic, involved actors, recording conditions, genre, viewers scores, etc. The net worth of these informative labels is large, specially taking into account that they allow the interconnection of information among documents. This concept of interconnection is very helpful for both pro-

[1] www.ncta.com
[2] www.fcc.gov
[3] www.netflix.com/en/about-netflix
[4] www.youtube.com/en/press

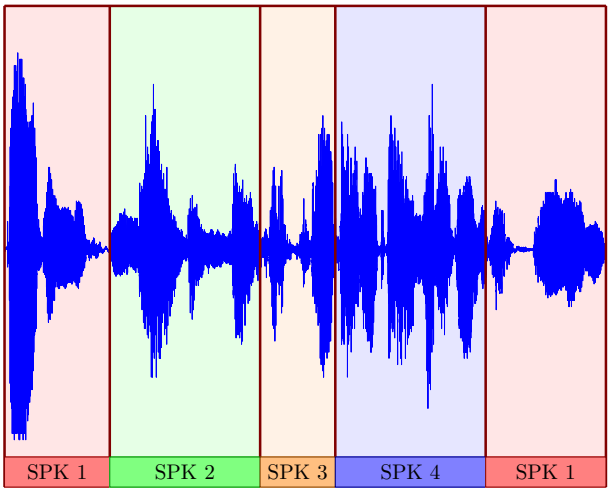


Figure 1.1: Example of diarization results. The speech from each individual speaker is differentiated.

fessionals and amateurs. An example of a professional environment is a newsroom, where properly indexed speech (speaker and transcription labels) makes possible the search for multiple speakers talking about the same topic, and vice versa, different contributions from the same speaker. In an amateur scenario, the worth of labels helps people with disabilities to enjoy the TV contents or suggests alternative TV series in terms of our preferences.

Unfortunately, the generation of these labels is not free of charge. While some of these labels may be easily provided by the content producer (genre, involved actors, etc.), others may imply an extensive analysis of the content (transcription), or even a thoughtful study about our preferences. Therefore, some of these labels may require large amounts of time and effort to be properly obtained. Sometimes real time related periods. These time limitations, as well as the increasing amount of data to tackle, have encouraged the development of automatic techniques for the estimation of these labels.

Diarization is the part of speech technologies dedicated to the differentiation of speakers in a recording. Given an audio with multiple speakers on it, diarization goal is the correct labeling of the audio in terms of the active speaker. Hence all the speech generated by each speaker is marked with a single speaker label. These speaker labels can be generic, without information about the speaker true identity. Nevertheless, its own deductive capabilities may include these identification properties. Traditionally, diarization has been described by the question "Who Spoke When?". An example of the diarization goal can be seen in Fig. 1.1

1.1 Motivation of the work

The motivation of this work is the development of diarization techniques in adverse conditions, paying special attention to broadcast data. Originally demanded as a support system for speaker recognition in telephone channel [Przybocki and Martin, 2004], diarization solutions have noticeably improved commercial purposes. However, some of these successful algorithms exploit restrictions of telephone audio, e.g. 2 speakers per conversation, making them domain dependent. Therefore, any proposed alternative strategy should maintain the accuracy levels without relying on these domain limitations.

In order to develop such approaches, we opt for the study of diarization techniques in the broadcast domain. This type of data, when collected *in the wild*, hardly ever presents the previously mentioned restrictions while containing a great diversity of acoustic conditions along its genres and shows. In fact, broadcast domain can be considered as a set of several subdomains, the TV shows, each one with its own particular characteristics. In consequence, our goal is the development of a single diarization system robust enough to deal with all these TV programs. In order to do so diarization must deal with a set of individual independent challenges, all of them responsible for great improvements in the final performance.

The first challenge is the clustering task, i.e. the grouping of the audio according to its active speaker. This task is a nondeterministic polynomial (NP) sort of problem with a large number of possible solutions, usually intractable in real life. Due to this intractability multiple suboptimal solutions to find the best labels can be proposed.

Another ordeal is the characterization of the speakers. Traditional state-of-the-art speaker characterization is accurate as long as training and evaluation conditions perfectly match. However, when some differences arise, such as language, domain, etc., these representations start to significantly fail. Moreover, while representations gain robustness as long as the amount of speech increases, diarization task tends to work with very short utterances, suffering from incomplete information. Hence improvements in this area compensating these undesired variabilities would cause relevant improvements in both speaker recognition and diarization as well.

Finally, the last challenge is the portability problem. Despite the quality of the considered speaker representations, part of the domain information cannot be totally compensated during their estimation, so domain mismatch issues still happen. In consequence, methods to deal with this situation must be considered. These techniques should allow the evaluation of some data by means of models trained with an out-of-domain data pool. This issue is extraordinarily important in multimedia information such as broadcast audio, where the large number of possible domains, i.e. TV shows, makes very difficult the creation of a general model perfectly adapted

to each domain.

1.2 Objectives and Methodology

The objectives of this thesis are the improvement of diarization capabilities so that systems could withstand the harmful conditions of the broadcast domain. These evolutions should be integrated in a single system, robust enough to deal with any sort of audio from the studied environment. Therefore, we should analyze possible evolutions in the previously described three lines of research.

Regarding to the speaker characterization problem, we want to improve the extraction of the **speaker representations**, obtaining efficient and discriminative characterizations of the involved speakers. Thus, we first seek a deeper understanding about the state-of-the-art modelling techniques based on subspace projection. Once this knowledge is is acquired, it will let us explore the limitations for these technologies, as well as propose new approaches designed accordingly.

With respect to the grouping task, our goal is the improvement of the **clustering techniques** estimating the diarization partitions. For this purpose, we make use of subspace-based techniques, specially PLDA, exploring different architectures and strategies.

Finally, we also must deal with the domain variability. In this area we will try to provide tools and strategies capable of decreasing the degradation of domain mismatch in circumstances where in-domain data is scarce or unavailable. In order to reach this goal we will deal with the **domain adaptation** problem by exploring the inference of unsupervisedly-crafted pseudo-speaker labels, obtained from the audio to diarize. These labels should be later used to specifically adapt the out-of-domain model to the evaluation audio.

1.3 Thesis organization

The outline of this thesis is very oriented to the different challenges we previously described. For this reason, this work is divided in five main parts, as shown in the conceptual map in Fig. 1.2:

- **Basic Knowledge**: This part is dedicated to present the diarization problem and an overview of the already proposed techniques in the state of the art (Chapter 2). Moreover, this part also starts the experimental activity, analyzing the characteristics of the broadcast domain and the performance of a baseline diarization system (Chapter 3).

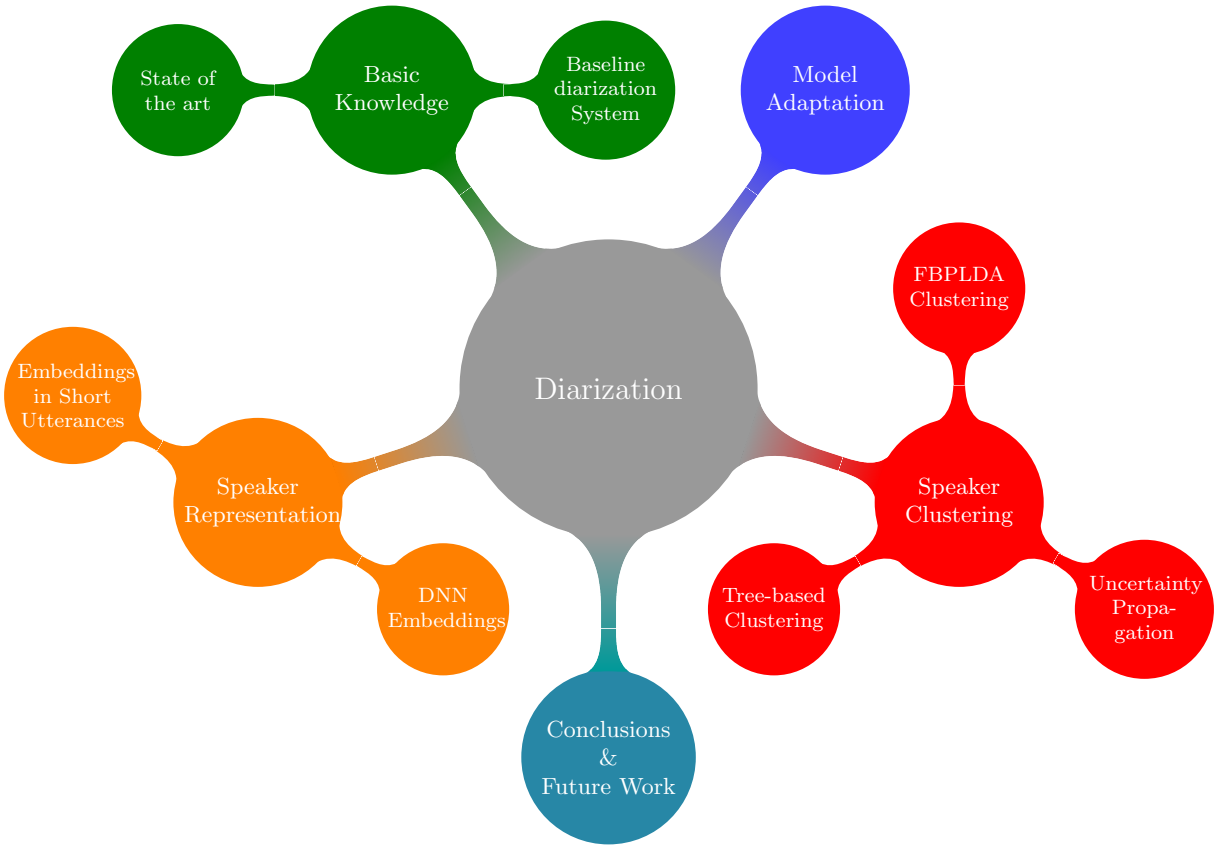


Figure 1.2: Conceptual map of the studied topics in this Thesis

- **Speaker Clustering**: This part is focused on the different tools to improve the performance of the clustering stage. First, we analyze the performance of the Fully Bayesian Probabilistic Linear Discriminant Analysis (FBPLDA) model, dealing with its weaknesses (Chapter 4). Chapter 5 updates the FBPLDA mode including the concept of Uncertainty Propagation (FBPLDAUP). Finally, in Chapter 6 we present a totally independent clustering solution by means of a tree-based approach.

- **Speaker Representation** This part of the thesis pays attention to the way speaker information is extracted from an audio utterance and compacted into a condensed representation, the embedding. First, we study the standard approximation for this information extraction, analyzing its impact on short utterances (Chapter 7). Later on, we make use of the learnt conclusions, applying them on the obtention of DNN-based embeddings for diarization (Chapter 8).

- **Model Adaptation**: This part, consisting on Chapter 9, works on the unsupervised extraction of in-domain information, suitable for the adaptation of out-domain labels. This

- **Summary**: This final part summarizes the conclusions for all the different parts of the thesis and proposes how this research could be followed in the future (Chapter 10)

Part I

Diarization Basic Knowledge

Chapter 2

Diarization State of the Art

The objective of this chapter is the revision of the state of the art in diarization. For this purpose, we take into account important reviews such as [Anguera et al., 2012][Tranter and Reynolds, 2006]. Our first goal is the identification of the main domains in which diarization has been applied. This differentiation helps understanding the evolution of diarization technologies. This knowledge allows the introduction of the two main approximations for diarization. Then, we explain in detail the functional blocks for the most popular diarization approach. Finally, the last part of the chapter includes a review about how to measure diarization performance.

2.1 Introduction

The diarization task includes all the techniques and procedures needed to differentiate the contributions of speakers given an audio. In the most general case, diarization works in an unsupervised way, i.e. without prior knowledge about the involved speakers nor its number. However, diarization can get benefited by means of the knowledge of these characteristics, usually simplifying the problem. Historically, diarization research has focused on three main domains of interest:

- **Telephone channel domain**. This environment involves the analysis of telephone conversations, characterized by the presence of few speakers, usually two, and conversational speech with short interventions. Moreover, telephone context usually considers close-to-mouth microphones and restricted *a priori* known channel conditions.
- **Broadcast domain**. This condition includes audios from mass media broadcasters (TV, radio, VoD, etc.). The most important feature in broadcast data is the large variability

of conditions. The variability in the number of speakers is almost unrestricted: from 3-4 up to 100 different speakers per hour of content, depending on the show. There is also variability in the type of speech: while some shows contain more read speech, e.g. the news, others hardly ever include it, being mainly composed of conversational speech, such as talk-shows. This characteristic has great relevance in diarization due to the length of the interventions. Whilst conversational speech usually consists of short interventions in order to maintain the conversation flow, read speech can generate longer turns due to the absence of feedback. Moreover, broadcast audio also presents variability of acoustic scenarios, such as studio and outdoors, each one with its own acoustic characteristics. Finally, except for live content, speech signal usually maintains high Signal to Noise Ratio (SNR), although very often speech is partially occluded by complementary acoustic additions such as music, and noises like canned laughter and applauses.

- **Meetings domain**. This scenario implies recordings from meeting rooms, where an undetermined number of people is recorded from one or multiple microphones. Hence, recordings from this domain mainly include conversational speech. In this domain recording conditions are also very relevant. Despite the fact that close-to-mouth microphones can be used, more often omnidirectional microphone arrays are considered. These arrays can be located in a single point, e.g. on top of the conference table, or spread along the room. Regardless of the microphone locations, the distance between speaker and microphone cannot be ignored. This distance is responsible for noticeable channel effects in the speech propagation up to the microphones, including degradations as reverberation. Besides, the stationarity of this transmission channel cannot be guaranteed, affected by the relative movements between speaker and microphone. Finally, these channel effects usually imply power losses of the signal, making speech quality more sensitive to noises.

The historic evolution of diarization originally started in the telephone domain. Due to its characteristics this domain provided the most restricted version of the diarization problem. Besides, there was a great interest for diarization solutions included in speaker recognition applications. This is why since 1996 diarization was part of NIST SRE evaluations [Przybocki and Martin, 2004]. Only after speaker recognition evolved its tools in terms of accuracy and robustness, diarization was able to export its knowledge to alternative domains, as in Rich Transcription (RT) evaluations [Garofolo et al., 2002], where alternative domains (broadcast news and meetings) complemented the conversational telephone speech.

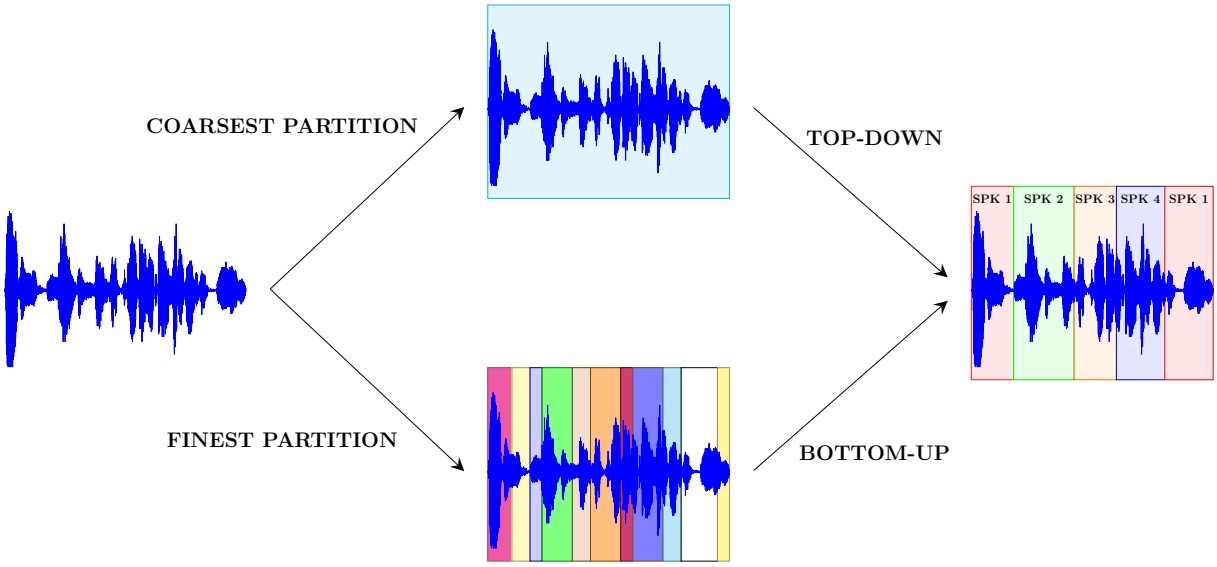


Figure 2.1: Schematic of Bottom-Up and Top-Down diarization

2.2 Main diarization strategies

Along literature several options have been proposed for the obtention of the diarization labels. However, most of these contributions can be grouped into two main conceptual approaches: Bottom-Up and Top-Down diarization strategies. Fig. 2.1 illustrates both diarization approaches in order to obtain the same diarization labels.

- **Bottom-Up**. The given audio is first divided into individual segments, in which a single speaker is assumed to be present. Then, these segments are clustered so all blocks from the same speaker are tagged with the same label.

- **Top-Down**. This alternative considers the opposite starting point. This approach starts considering a single speaker responsible of all the audio. Afterwards, the initial cluster is divided trying to match each final cluster with a real speaker in the audio.

In spite of their opposite approach, both strategies need to solve the same two challenges: Determining whether some part of the audio contains speech from a single speaker and finding the boundaries if necessary. Despite the apparent simplicity of both tasks, their development for real applications has required several contributions in the literature. Nevertheless, both tasks are still far for being totally solved.

Despite both Bottom-Up and Top-Down approaches are equally valid, they are not similarly popular. While both options have been developed along multiple publications, in recent years

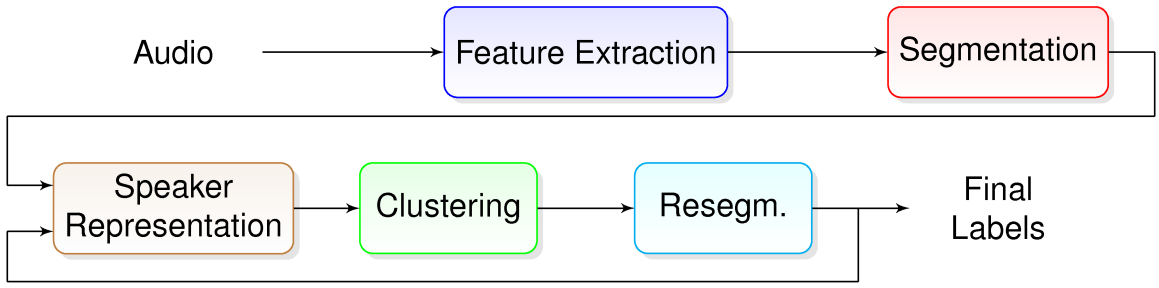


Figure 2.2: General schematic for a diarization system

the Bottom-Up strategy has gained much more awareness than the Top-Down counterpart. A reason for this popularity is the fit among the latest improvements in speaker recognition and the Bottom-Up diarization pipeline, making their inclusion straightforward. Under these circumstances, Bottom-Up diarization has taken its performance to unprecedent levels of quality. In consequence, this option has recently gained popularity becoming the standard diarization approach nowadays.

2.2.1 Bottom-Up diarization systems

The popularity of Bottom-Up diarization has inspired the development of a standard architecture, which we present in Fig. 2.2. This schematic describes the standard considered blocks to transform the input raw audio into the desired final labels. The functionality of each block is described as follows:

- **Acoustic Feature Extraction**. Raw speech audio is a very complex signal with many sorts of information. While some of them are valuable depending on the application (speaker, speech, language, etc.), others are not of interest (channel, noises, etc.) because they can alter our estimates. The feature extraction step aims to transform the raw signal into a faithful but compact representation of the acoustic information, simplifying the access to our target information and compensating those harmful degradations.

- **Segmentation**. Generally speaking, segmentation is the task of dividing an audio into pieces according to an attribute, which should remain homogeneous along the total length of each piece. Focusing on diarization, the division attribute is the speaker identity. Thus, the goal of diarization segmentation is the division of a given audio into segments where a single speaker is present in them. This system must exploit the homogeneity of data in short periods of time to find the boundaries between speakers. An ideal segmentation step should provide the time marks for the different speaker interventions in an audio.

- **Speaker Characterization**. Speaker characterization is a high-level information extraction which collects the speaker information from the acoustic features. In order to properly do so, it requires working with audio from a single speaker. Thanks to this requirement, highly evolved techniques work along the given input segments, enhancing their speaker discriminative properties while compensating the harmful variabilities. Moreover, this process usually converts variable-length segments into fixed-dimension compact representations, more suitable for postprocessing.

- **Clustering**. The output of the segmentation step is a set of acoustic fragments with a single speaker in each of them. However, the same speaker may have produced more than one segment. The clustering stage is responsible for grouping all those segments from the same speaker and label them with a unique tag. For this purpose, clustering takes the segment representations as input, generating the diarization labels as output.

- **Resegmentation**. Resegmentation is an optional extra segmentation step to refine the initial segmentation boundaries. This extra border tuning takes advantage of the inferred clustering output, with an accurate knowledge about the evaluation audio. Resegmentation output may be considered as diarization labels or be fedback into the system for further refining.

2.3 Acoustic features for diarization

In order to differentiate speakers, diarization systems require a subsystem capable of providing discriminative characteristics at each time step of an audio. These characteristics, also known as features, should maximize their classification capabilities. For this reason, they try to represent the audio information in a tractable manner, simplifying the information gathering while reducing harmful sorts of variability (noise, channel information, etc.) meanwhile. Moreover, feature extraction is the first diarization block, hence no assumption like number of speakers nor their identity, speaker transitions, etc. can be done.

The most popular features so far are those commonly known as short-term acoustic features. Originally designed for speech recognition, these features carry out a spectral analysis of the raw signal while inspired by both the human production and perception systems. Because the speech signal is not stationary, this analysis must be performed in short analysis windows.

The most popular features are the Mel Frequency Cepstral Coefficients (MFCCs), originally presented in [Davis and Mermelstein, 1980]. These features propose a short-time analysis of the

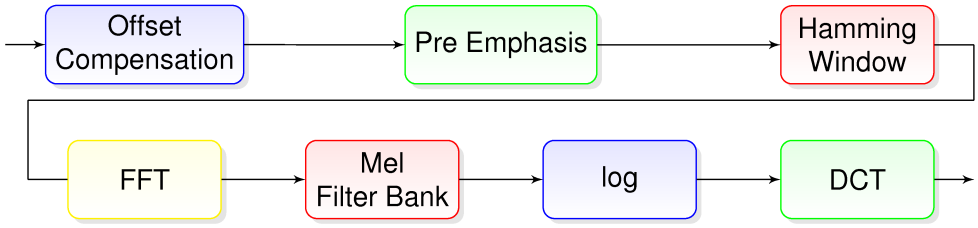


Figure 2.3: Schematic of the MFCC extraction pipeline

speech audio. Prior to this analysis, the audio signal undergoes some conditioning steps (offset compensation, pre-emphasis and Hamming windowing). Then, for each resulting window a cepstral analysis is carried out. In the process, a non-uniform filter bank describes the analysis bands. This filter bank, based on the non-linear Mel frequency scale, imitates the human acoustic response. Finally, the log-spectral information is decorrelated by means of the Discrete Cosine Transform. Its full schematic is shown in Fig. 2.3

A popular alternative to MFCCs are the Perceptual Linear Predictive (PLP) features [Hermansky, 1990]. This approach computes the LPC coefficients of short-time windows and then transform them to LPC-cepstrum.

In recent times another type of feature has become popular for speaker recognition, the Constant Q Cepstral Coefficients [Todisco et al., 2017]. These features propose an alternative frequency analysis in which frequency bands are analyzed with filters of constant quality factor Q. This choice reassures a constant relationship between the central frequency and the bandwidth for all the involved filters. Thus, lower frequencies gain extra frequency resolution while high frequencies obtain more time resolution.

Despite the good performance of the acoustic features, their original purpose was speech recognition, where speaker information should be compensated. Thus, alternative characteristics were proposed. In [Yamaguchi et al., 2005] a set of features is proposed, such as energy, pitch frequency, peak-frequency centroid and peak-frequency bandwidth. These features are complemented in [Huang and Hansen, 2006], where Perceptual Minimum Variance Distortionless Response (PMVDR), Smoothed Zero Crossing Rate (SZCR) and Filter-Bank Linear Coefficients (FBLC) were presented. Although the mentioned features succeeded in overcoming the MFCC performance, their benefits were not significant enough to replace them. Moreover, prosodic features have also been taken into consideration. In [Shriberg et al., 2005][Friedland et al., 2009], some features based on prosody have been studied, sometimes combined with traditional MFCCs. The joint work again provides small benefits but not significant enough. Thus, prosody contains useful speaker information, but we still do not know how to exploit it.

The advent of Deep Neural Networks (DNNs) in speech recognition has also contributed with new options. In [Zhang et al., 2014] Bottleneck Features (BNFs) are proposed for speaker recognition tasks. These features are the result of non-linear transformations on traditional features, such as MFCCs. These transformations are not manually crafted, but learnt when a senone recognition DNN is trained. BNFs are extracted during the forward step of the DNN, in a low dimension layer known as bottleneck. These features clearly outperformed MFCCs becoming state of the art. Some results indicate that both MFCCs and BNFs could jointly work [Lozano-diez et al., 2016] [Viñals et al., 2016]. However, posterior changes in the speaker recognition paradigm, specially speaker characterization, made BNFs unnecessary.

Regardless of the feature nature, all of them contain the target speaker information, as well as other harmful variabilities, such as channel distortions. Multiple normalization techniques have been proposed in speaker recognition in order to compensate the channel variability in the feature space.

Thanks to the cepstral concept in MFCCs, those channel distortions constant over time with convolutional nature become additive effects in the cepstral domain. Therefore, when these distortions are time invariant they can be removed by the Cepstral Mean Subtraction (CMS), proposed in [Bimbot et al., 2004]. This CMS concept is extended in the Cepstral Mean and Variance Normalization (CMVN) [Alam et al., 2011], in which features are normalized in both mean and variance. Other alternatives propose Feature Warping [Pelecanos and Sridharan, 2001], a non-linear transformation that transforms the feature distribution to fit into a Standard Normal.

However, some studies such as [Kenny et al., 2010] illustrate that speaker diarization systems can obtain better results when unnormalized features are taken into account. Thus, according to this result, channel effects may be informative for speaker discrimination.

2.4 Audio segmentation

Segmentation of audio is the task of dividing an audio according to its multiple sources. This division is done by finding those time marks between which the active source remains steady. Moreover, segmentation should also classify the active source at each segment if possible. Depending on how this division is done, we can distinguish between "Segmentation & Classification" (S&C) and "Segmentation by Classification" (SbC). The former option first divides the audio into segments, which are later classified among the candidate sources. By contrast, the latter classifies small segments of time, even frame level classifications, obtaining the segmentation by composition of the classification labels. Whereas S&C does not require *a priori*

information about the classes, SbC does for the short time classification.

The segmentation problem is also known by alternative names when certain sources must be distinguished. Some examples are Voice Activity Detection (VAD) determining when somebody is speaking, Speaker Change Point Detection (SCPD) finding speaker turns, or even diarization itself. In fact, diarization is a segmentation task in which speakers are the sources to differentiate. This is why diarization is sometimes referred as segmentation in the literature.

In the context of Bottom-Up diarization, segmentation is the task intended to isolate contiguous segments of audio with a single speaker in them. Therefore, this task must identify when somebody is talking and find those borders where the active speaker changes. Thus, segmentation must cover the transitions between speakers, but also the turn from speech to non-speech and vice versa. Then, segmentation can be divided into two different segmentation subtasks:

- **Voice Activity Detection (VAD)**: This segmentation task must find the speech non-speech transitions. Besides, VAD must classify each audio segment.
- **Speaker Change Point Detection (SCPD)**: This segmentation seeks identifying those boundaries between speakers. This segmentation usually works on top of VAD labels, assuming to exclusively work with speech data.

The strategies to carry out these two segmentations are very different. VAD must deal with a closed set of two *a priori* known classes, speech and non-speech. Therefore, supervised models can be trained for each class. These models make possible both the SbC approach as well as the S&C strategy. By contrast, SCPD usually lacks from this sort of *a priori* information. For those cases we can only identify the speaker boundaries. Consequently, as shown in [Chen and Gopalakrishnan, 1998], SCPD has taken into consideration several strategies to achieve its goal:

- **Metric-Based Segmentation**. A metric is computed for a window of audio, divided into two parts by a hypothetical speaker boundary. This metric measures how better the data in the window are modeled in case the candidate boundary is true compared to the same scenario without any speaker turn. Only those hypothetical boundaries whose metric overcomes a threshold are considered as final borders.
- **Model-Based Segmentation**. Speaker models are trained to measure how likely a certain audio is generated by each speaker. According to this measure, the audio is classified among all candidate speakers. Each candidate speaker must have its trained model. Therefore, prior information about the voice characteristics from every speaker must be available for model training.

- **Silence-Based Segmentation**. Any speaker turn is always between two segments of non-speech. Thus, VAD transitions become hypothetical speaker borders as well.

In general, SCPD in diarization systems combines the three types of segmentation. Metric-based segmentations are suitable for the initial segmentation, where no further information is required. This segmentation can be complemented with silence-based segmentations, so VAD boundaries are taken into account. Finally, model-based segmentations can be applied in posterior steps when obtained diarization labels can be used as prior information to train the models.

2.4.1 Metric-based segmentation

Metric-based segmentation is the most typical approach to the SCPD problem. Its simplicity and the independence from prior information provides flexibility to deal with any scenario, gaining great popularity for this matter.

Metric based segmentation works as a hypothesis test for short windows of analysis, around 3-5 seconds. Considering the analysis window as an ordered set of N frames $O = \{o_1, ..., o_n, ..., o_N\}$, we hypothesize the existence of a single frame b that divides the whole set into two subsets, $O_L(b)$ and $O_R(b)$, generated by the speakers L and R respectively. Hence the two hypotheses taken into consideration are: On the one hand there is no such sample b within O because a single speaker is responsible for all the frames within the set (hypothesis H_0). On the other hand the frame b is in fact a Speaker Change Point (SCP) separating the subsets from speakers L and R. The hypothesis test is done by means of the function $D(O_L(b), O_R(b))$, a metric estimating how better O is represented by hypothesis H_1 rather than by hypothesis H_0. This metric is also referred as distance because it measures the extra representativeness of hypothesis H_1 with respect to hypothesis H_0. The final decision is made by comparing the measured value with a threshold th.

$$D(O_L(b), O_R(b)) \underset{H_0}{\overset{H_1}{\gtrless}} th \tag{2.1}$$

The previous test only studies an analysis window for a specific hypothesis boundary b. However, the location of this boundary, if real, is unknown. Therefore, within an analysis window a set of candidate boundaries must be considered, ideally all samples. Nevertheless, due to our initial assumption there is at maximum one boundary along the analysis window only the candidate boundary with highest distance should be compared with the threshold.

The previous test is only applicable to short analysis windows. When audios become longer, strategies to extrapolate the window analysis concept are required. A solution is based on an

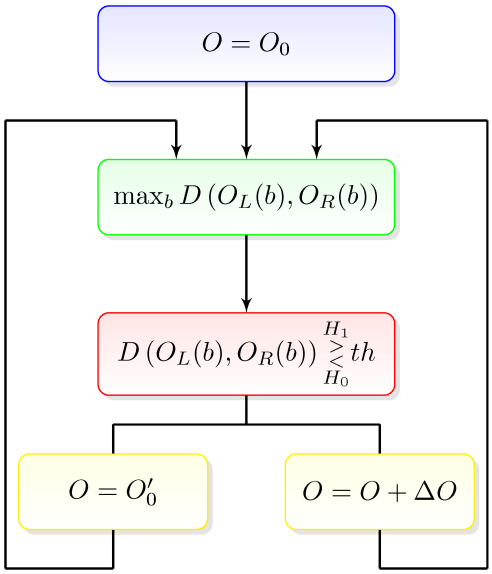


Figure 2.4: Scheme for a sliding window metric based segmentation

incremental sliding window analysis, acting as follows: The analysis of a certain window must decide if this portion of audio contains one (H_0) speaker or two (H_1). If opted for hypothesis H_0, we can expand the analysis window, increasing its length with an extra portion ΔO, and repeat the analysis. By contrast, when choosing H_1 we have assumed the frame b to be a border within the window. Thus, a new speaker starts after the border frame b. In consequence we shift the following analysis window O' to start at the border b. In this case the size of the new analysis window is reset to the minimum size. Fig. 2.4 illustrates this procedure.

The previously explained procedure is insensitive to the metric $D(O_L(b), O_R(b))$. In the literature many alternative metrics have been proposed. Some examples are:

2.4.1.1 Bayesian Information Criterion (BIC)

Bayesian Information Criterion (BIC) [Schwarz, 1978] is an information metric, i.e. a method to measure how well the model $\mathcal{M}$ represents some data set $O = \{o_1, ..., o_n, ..., o_N\}$ of N samples. The purpose of BIC is the comparison among models according to how well they fit some given data. For this reason, BIC makes use of the likelihood of the data given the model but penalized by an extra term. This penalization term must compensate the model complexity for fair comparisons. Therefore, BIC is defined as:

$$BIC(\mathcal{M}) = \ln P\left(O|\mathcal{M}\right) - \lambda\frac{1}{2}\#(\mathcal{M})\log(N) \tag{2.2}$$

where $\ln P(O|\mathcal{M})$ represents the log-likelihood of the data given the model, $\#(\mathcal{M})$ is the number of tunable parameters in the model $\mathcal{M}$ and λ is a finetuning parameter.

For segmentation matters, BIC can be applied when comparing two hypotheses. In [Chen and Gopalakrishnan, 1998] ΔBIC is proposed to explain the content in the analysis window O with a hypothetical border b: On the one hand, a model $\mathcal{M}_U$ assumes a single speaker within the whole window (hypothesis H_0). On the other hand, model $\mathcal{M}_{LR}(b)$ considers two speakers, L and R respectively, separated at time b (hypothesis H_1). Hence ΔBIC is defined as:

$$\Delta BIC(H_1, H_0) = BIC(H_1) - BIC(H_0) = R(H_1, H_0) - \lambda P \tag{2.3}$$

where $R(H_1, H_0)$ represents the log-likelihood ratio (LLR) between hypotheses H_1 and H_0, and P is the excess complexity term of H_1 with respect to H_0 to compensate.

Among multiple distributions, the Gaussian distribution with full rank covariance matrix Σ has gained awareness when combined with ΔBIC. When speakers are modeled according to this distribution, the formulation for ΔBIC becomes very compact. By working with Gaussian distributions $R(H_1, H_0)$ is simplified to:

$$R(H_1, H_0) = \frac{N_U}{2}\ln(|\Sigma_U|) - \frac{N_L(b)}{2}\ln(|\Sigma_L(b)|) - \frac{N_R(b)}{2}\ln(|\Sigma_R(b)|) \tag{2.4}$$

where $N_L(b)$ and $N_R(b)$ are the number of samples at each side of the candidate border b, and N_U is the length of the analysis window. Σ_U, $\Sigma_L(b)$ and $\Sigma_R(b)$ are the covariance matrices estimated with the data in the whole window O and its partitions $O_L(b)$ and $O_R(b)$ respectively.

The benefits of the Gaussian distribution also reach the penalty term P, now computed as:

$$P = \frac{1}{2}(d + \frac{1}{2}d(d+1))log(N_U) \tag{2.5}$$

where d is the dimension of the Gaussian distribution, equal to the data dimension.

Despite the benefits of the Gaussian distribution, it is not the only alternative. For instance, other proposals dislike the tunable hyperparameter λ. According to the general definition, this hyperparameter can be removed if there are no extra modeling capabilities between hypotheses H_0 and H_1. In [Ajmera and Wooters, 2003] this condition is achieved by considering GMMs to model each speaker, using the same number of components for both hypotheses.

2.4.1.2 Kullback-Leibler Divergence (KL)

The Kullback-Leibler divergence [Kullback and Leibler, 1951] is also a popular measurement to determine how much a given distribution differs from its reference. For discrete probability distributions P and Q, the Kullback Leibler divergence between P and Q is defined as:

$$D_{\mathrm{KL}}(P||Q) = \sum_{x} P(x) \log \left(\frac{P(x)}{Q(x)} \right) \tag{2.6}$$

Unfortunately, its original definition is not symmetric, i.e., the KL divergence of Q with respect to P ($D_{\mathrm{KL}}(P||Q)$) may not be the same as the divergence of P with respect to Q ($D_{\mathrm{KL}}(Q||P)$). In consequence a symmetrized version, known as KL2 divergence, is used instead. This divergence for distributions P and Q is defined as:

$$D_{\mathrm{KL2}}(P||Q) = D_{\mathrm{KL}}(P||Q) + D_{\mathrm{KL}}(Q||P) \tag{2.7}$$

Moving to diarization, this distribution is considered in segmentation in [Siegler et al., 1997] [Delacourt and Wellekens, 2000].

2.4.1.3 Deep Neural Networks (DNNs)

Thanks to the evolution of neural networks many of the tasks previously carried out by other means, such as statistics, are now performed by this technology. Regarding segmentation, some contributions have attempted the inclusion of DNNs in this task.

In [Gupta, 2015] DNNs are used as classifiers. The hypothetical boundary frame is stacked along its context window, feeding a monolithic DNN consisting of feed forward layers. The final layer classifies the boundary frame as real or not. Moreover, a likelihood measure can be obtained in the process.

By contrast, DNN regression capabilities can also been applied. In [Hruz and Zajic, 2017] the neural network must carry out the regression of the transition probability, softened during training. For this purpose, input data is treated by means of stacks of convolutional neural networks.

In both cases, DNNs work as standalone systems. However, both architectures fit the given more general definition, where a neural network provides a metric for a fixed-length analysis window and compared against a threshold.

2.4.2 Model-based segmentation

Despite the fact that metric-based segmentations are the most popular ones, other alternatives have also been proposed. Considering model-based segmentations, [Li et al., 2009] considers Hidden Markov Models (HMMs) for segmentation. The model represents each class by means of a 64-Gaussian GMM. This concept is evolved in [Diez et al., 2018], where classes are represented with tied GMMs, more suitable for speaker representation.

Finally, some systems [Garcia-Romero et al., 2017][Diez et al., 2019] work in terms of a coarse SbC approach. They prefer working with very short fixed-length (around 1.5 seconds) segments, not taking care for boundaries. These systems rely on the latest speaker characterization techniques, which have evolved to provide robust enough representations when working with very short segments. By doing so, they alleviate the computational costs while only introducing a small proportion of corrupted segments: as many degraded segments as real boundaries. Besides, these systems usually count with resegmentation systems to eliminate the generated distortions once speaker models are available.

2.5 Speaker characterization

The nature of speech makes this information to have a sequential nature. Human beings concatenate multiple sounds to transmit the desired information. However, there is no limitation in terms of its length nor the message. It can either be a large speech or a short reply to a closed question, i.e., "yes" or "no". Moreover, it can include all the acoustic units or just a restricted set. Speaker recognition technologies should provide a tool to robustly encode the identity of the involved speaker regardless of the intra-speaker variability, i.e. the variability within all the possible utterances from the same speaker. Some reviews such as [Furui, 2004][Kinnunen and Li, 2010] provide a good overview about the evolution of these technologies.

2.5.1 Early days

Some of the first successful speaker recognition systems were based on the correlation of spectrograms [Pruzansky, 1963]. This idea was later evolved to take into account the formant analysis [Doddington, 1971]. Because these techniques were not powerful enough to deal with text-independent recognition, some alternatives were explored for the following decade. Some proposals during those years are the instantaneous spectra covariance matrix [Li and Hughes, 1974], spectrum and fundamental frequency histograms [Beek et al., 1977] or linear prediction coefficients [Sambur, 1972]. The following great evolution appeared with the consideration of template models: Dynamic Time Warping (DTW) [Furui, 1981] and Vector Quantization (VQ) [Rosenberg and Soong, 1987] [Soong et al., 1985], which proposes short time feature vectors compressed in codebooks. This principle was later evolved as long as matrix quantiziers for multi-frame were also proposed [Juang, 1990].

2.5.2 Model-based representations

In the 80s, a great evolution in the characterization philosophy was proposed. Rather than considering speech as a deterministic process where features could be measured, state-of-the-art contributions started to define statistical models as generators for speech. Moreover, these generators were often designed only taking into account the acoustic information, not considering high-level crafted features.

The generative sort of solution has many advantages. First, all segments are supposedly generated by a known distribution, a parametric solution perfectly described by a closed set of parameters, some of them speaker dependent. Besides, this sort of solution allows the same model to work with variable-length segments while providing a fixed-dimension speaker representation. Finally, statistical solutions can also provide protection against different types of randomness associated with the voice (phonetic variability, noises, etc.).

When choosing the distribution to better represent speakers, Gaussian distributions are usually taken into account. Very well known among statisticians, Gaussian distributions have worthy properties. However, Gaussian distributions are too simple to properly represent all the variability and conditions in speech. Thus, combinations of them, Gaussian Mixture Models (GMMs) are considered instead. This approach is considered under the assumption that a linear combination of enough Gaussians should be able to reproduce any distribution.

2.5.2.1 Gaussian Mixture Models (GMMs)

Gaussian Mixture Models are generative statistical models first introduced in speaker recognition in [Reynolds and Rose, 1995]. They are composed by the weighted sum of C Gaussian components, each one with its own weight π_c, mean vector $\boldsymbol{\mu}_c$ and covariance matrix $\boldsymbol{\Sigma}_c$, being $c = 1..C$. Thus, the sequence $\boldsymbol{O} = \{\boldsymbol{o}_1, ..., \boldsymbol{o}_n, ..., \boldsymbol{o}_N\}$ generated by a GMM has been randomly drawn as:

$$P\left(\boldsymbol{O}|\mathcal{M}\right) = \prod_{n=1}^{N} \sum_{c}^{C} \pi_c \mathcal{N}\left(\boldsymbol{o}_n|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c\right) \tag{2.8}$$

The evaluation of these systems worked in terms of a loglikelihood ratio. Two loglikelihood terms were considered, both taking into account the test audio $audio_{\text{test}}$ but considering two different models: A model of the claimed enroll speaker ($\mathcal{M}_{\text{enroll}}$) and a model representing speakers except for our enrollment one ($\mathcal{M}_{\overline{\text{enroll}}}$).

$$\text{llr} = \ln\left(\frac{P\left(audio_{\text{test}}|\mathcal{M}_{\text{enroll}}\right)}{P\left(audio_{\text{test}}|\mathcal{M}_{\overline{\text{enroll}}}\right)}\right) \tag{2.9}$$

While $\mathcal{M}_{\text{enroll}}$ was straightforward, the definition of $\mathcal{M}_{\overline{\text{enroll}}}$ was not so clear. Many systems worked with a pool of cohort models, chosen for each trial according to different criteria.

Then, this idea was evolved in [Reynolds et al., 2000], which proposes the GMM-UBM paradigm. First, this contribution integrates the cohort of alternative speakers into a single model, responsible to represent the total variability of the acoustic data. This general model, a large GMM trained with several speakers, is known as Universal Background Model or UBM. Furthermore, instead of building from scratch individual enroll models $\mathcal{M}_{\text{enroll}}$, it proposes the option of their construction as a MAP adaptation from the UBM, specifically an adaptation of the component means. The obtained benefits are a tighter coupling between models, and faster scoring techniques. In this scenario, the proposed llr was:

$$\text{llr} = \ln\left(\frac{P\left(audio_{\text{test}}|\mathcal{M}_{\text{enroll}}\right)}{P\left(audio_{\text{test}}|\mathcal{M}_{\text{UBM}}\right)}\right) \tag{2.10}$$

2.5.2.2 Support Vector Machines (SVM)

The GMM-UBM strategy became a milestone in speaker verification, specially regarding the way to model speakers. However, alternative scoring approaches were attempted. Within this line of research Support Vector Machines (SVMs) were proposed for speaker recognition [Campbell et al., 2006a], leading to the SVM-GMM strategy.

Support Vector Machines are binary classifiers that project the input data into a high-dimensional space where a hyperplane separates the two classes. The evaluation in SVMs is defined as follows:

$$f(x) = \sum_{c=1}^{C} y_c \alpha_c K(x_c, x) + b \tag{2.11}$$

where $f(x)$ stands for the distance of the utterance x with respect to the hyperplane. x_c, α_c and y_c represent the support vectors, weights and labels respectively, with the restriction that $\sum_{c=1}^{C} y_c \alpha_c = 0$ and $\alpha_c > 0$. The labels y_c take the value $+1$ for one class and -1 for the other one. Besides b represents the hyperplane bias. Finally, $K(\cdot,\cdot)$ stands for the kernel function, responsible for projecting the data into the high-dimension space and calculating distances terms If $K(\cdot,\cdot)$ is restricted to satisfy the Mercer condition, the Kernel condition can be expressed as an inner product as:

$$K(x, y) =< g(x), g(y) > \tag{2.12}$$

where $g(\cdot)$ is the transformation into the highly dimensional space.

SVMs are trained by a maximum margin strategy. This type of training must identify a hyperplane which properly classifies the training elements while satisfying the following re-

striction: the chosen hyperplane must keep the maximum distance with respect to the training populations of both classes. This request forces the hyperplane to provide the maximum margin protection against spurious data during evaluation.

The inclusion of SVMs in speaker recognition [Campbell et al., 2006a] was carried out by proposing a kernel that bounds the KL divergence. In our scenario, KL divergence measures the distance between utterances $\boldsymbol{O}_a$ and $\boldsymbol{O}_b$, modeled by GMMs $\mathcal{M}_a$ and $\mathcal{M}_b$ respectively. Both GMMs are obtained according to the GMM-UBM paradigm, hence they share the component weights π_c and the component covariance matrices $\boldsymbol{\Sigma}_c$, only differing at the component means $\boldsymbol{\mu}_c$. The kernel accomplishing this request is:

$$K(\boldsymbol{O}_a, \boldsymbol{O}_b) = \sum_{c=1}^{C} \left(\sqrt{\pi_c}\boldsymbol{\Sigma}_c^{1/2}\boldsymbol{\mu}_c^a\right)\left(\sqrt{\pi_c}\boldsymbol{\Sigma}_c^{1/2}\boldsymbol{\mu}_c^b\right) \tag{2.13}$$

In consequence, the kernel function can be interpreted as the inner product of the two GMM supervectors, a concatenation of the GMM means undergoing a diagonal scaling. Applied to our previous definition of SVMs, the enrollment supervector constitutes the set of support vectors and the test supervector plays the role of evaluated utterance, deciding whether it comes from the enrollment speaker.

The GMM-SVM paradigm was complemented with the Nuisance Attribute Projection (NAP) concept. This idea, introduced in [Campbell et al., 2006b], considered the compensation of the intra-speaker variability present in the supervectors. This compensation is performed by estimating a low rank matrix $\mathbf{U}$, also known as eigen-channels matrix, which defines the intra-speaker variability within the supervector space. Once modeled, a matrix $\mathbf{P} = \mathbf{I} - \mathbf{U}\mathbf{U}^T$ can be introduced in the kernel function already seen in eq (2.13):

$$K(\boldsymbol{O}_a, \boldsymbol{O}_b) = \sum_{c=1}^{C} \left(\sqrt{\pi_c}\boldsymbol{\Sigma}_c^{1/2}\boldsymbol{\mu}_c^a\right)\mathbf{P}\left(\sqrt{\pi_c}\boldsymbol{\Sigma}_c^{1/2}\boldsymbol{\mu}_c^b\right) \tag{2.14}$$

2.5.2.3 Joint Factor Analysis (JFA)

Joint Factor Analysis (JFA) [Kenny, 2005] is an evolution of the GMM representations, paying special attention to two concepts developed with the GMM-SVM approach: supervectors and subspaces for certain variabilities. Taking both concepts into account the JFA methodology evolves de the GMM-UBM paradigm decomposing the adapted GMM supervector as a sum of terms:

$$\boldsymbol{\mu}_j = \boldsymbol{\mu}_{\text{UBM}} + \mathbf{V}\mathbf{y}_i + \mathbf{U}\mathbf{x}_k + \mathbf{D}\mathbf{z}_j \tag{2.15}$$

where $\boldsymbol{\mu}_j$ is the adapted supervector mean of utterance j. $\boldsymbol{\mu}_{\text{UBM}}$ represents the mean supervector from the UBM model. The term $\mathbf{V}\mathbf{y}_i$ is the speaker dependent term. $\mathbf{V}$ is a low rank matrix describing the subspace of the inter-speaker variability while $\mathbf{y}_i$ is a tied latent variable, i.e. a latent variable whose value is the same for all utterances from speaker i, responsible for the utterance. Similarly, we have the term $\mathbf{U}\mathbf{x}_k$ or channel term. $\mathbf{U}$ is a low rank matrix describing the channel variability space and $\mathbf{x}_k$ is the tied latent variable for all utterances with the same channel k, including utterance j. Finally, we have the term $\mathbf{D}\mathbf{z}_j$, which must explain the remaining variability. For this purpose, $\mathbf{D}$ is a diagonal matrix and $\mathbf{z}_j$ a latent variable unique for the utterance. All the three latent variables, $\mathbf{y}_i$, $\mathbf{x}_k$ and $\mathbf{z}_j$ are Standard Normal distributed.

2.5.3 Embedded representations

The following large evolution implied the improvement of the already proposed models, but also a new methodology. On the one hand, models including latent variables to map the speaker information significantly improved the performance. On the other hand, the approach of the GMM-SVM showed that information could be extracted from the models and independently treated. The combination of both ideas created the embedding paradigm, defining models that constrain the speaker information into a restricted space where a latent variable should explain each speaker. From these latent variables we could extract compact representations, voiceprints for each speaker, also known as embeddings.

Embeddings offer several advantages compared with previous approaches. Once embeddings are extracted, they can be decoupled from the original extraction method, simplifying their storage. Moreover, this decoupling makes impossible the return to the original audio, guaranteeing privacy. Finally, the obtained embeddings can be postprocessed by alternative methods, also known as backends. In fact, the current speaker recognition state of the art, from which most of these technologies are conceived, is dominated by the embedding-backend pipeline.

In the following lines some of the most popular embeddings are presented, and one of the most popular backends, the Probabilistic Linear Discriminant Analysis (PLDA) is explained afterwards.

2.5.3.1 I-vectors

I-vectors [Dehak et al., 2011] are a direct evolution of the JFA modeling. Rather than differentiating between speaker and channel factors, i-vector model integrates them into the total variability subspace. This fusion makes the latent variable store both speaker and channel information together. Moreover, latent variables are not linked among utterances anymore, being

only tied along the samples from the utterance. Besides, this model no longer considers a residual variability term. In consequence, the utterance j, consisting of the sequence of frames $O = \{o_1, ..., o_n, ..., o_N\}$, is now modeled by a GMM whose mean supervector $\boldsymbol{\mu}_j$ is defined as:

$$\boldsymbol{\mu}_j = \boldsymbol{\mu}_{\text{UBM}} + \mathbf{T}\mathbf{w}_j \tag{2.16}$$

where $\boldsymbol{\mu}_{\text{UBM}}$ again describes the UBM mean supervector. $\mathbf{T}$ stands for a low rank matrix describing the total variability subspace and $\mathbf{w}_j$ is the latent variable depending on the utterance.

The mentioned model still can be evaluated in terms of likelihoods as in JFA. Nevertheless, this technology evolved to become a voiceprint extractor. The commonly used i-vector is the mean of the posterior distribution of the latent $\mathbf{w}_j$ given the utterance j. This distribution is Gaussian and defined as:

$$\mathbf{w}_j \sim \mathcal{N}\left(\mathbf{w}_j | \boldsymbol{\mu}_{\mathbf{w}}, \Sigma_{\mathbf{w}}\right) = \mathcal{N}\left(\mathbf{w}_j | \mathbf{L}_{\mathbf{w}}^{-1}\Gamma_{\mathbf{w}}, \mathbf{L}_{\mathbf{w}}^{-1}\right) \tag{2.17}$$

$$\Gamma_{\mathbf{w}} = \sum_{c=1}^{C} \mathbf{T}_c^T \Sigma_c \sum_{n=1}^{N_j} \gamma_{nc}\left(\boldsymbol{o}_n - \boldsymbol{\mu}_c\right) = \sum_{c=1}^{C} \mathbf{T}_c^T \boldsymbol{\Sigma}_c \overline{\mathbf{F}}_c \tag{2.18}$$

$$\mathbf{L}_{\mathbf{w}} = \mathbf{I} + \sum_{c=1}^{C} \mathbf{T}_c^T \sum_{n=1}^{N_j} \gamma_{nc} \Sigma_c \mathbf{T}_c = \mathbf{I} + \sum_{c=1}^{C} \mathbf{T}_c^T N_c \boldsymbol{\Sigma}_c \mathbf{T}_c \tag{2.19}$$

where $\boldsymbol{\mu}_{\mathbf{w}}$ represents the mean of the posterior distribution and $\boldsymbol{\Sigma}_{\mathbf{w}}$ is its covariance. These terms are constructed in terms of $\mathbf{T}_c$, the submatrix from $\mathbf{T}$ describing the contribution of the cth Gaussian component, and $\boldsymbol{\Sigma}_c$, the covariance matrix for the cth component in the UBM. The information of the utterance is contained in N_c and $\overline{\mathbf{F}}_c$, the zeroth and centered first order Baum Welch statistics for the cth component respectively. Finally, N_j represents the total amount of samples in the utterance j. Both of them are obtained in terms of the responsibilities γ_{nc}, the probability of the nth sample o_n to be drawn from component c of the GMM-UBM.

2.5.3.2 Hybrid i-vectors

The latest great evolution of neural networks, affecting both software and hardware, has become a milestone along most artificial intelligence tasks. This evolution also reached speech technologies [Hinton et al., 2012], including speaker characterization. In these tasks, at first, this acquisition of the new approaches was smooth, complementing existing state-of-the-art technologies.

A proposed inclusion of DNNs in i-vectors was presented in [Lei et al., 2014] as hybrid i-vectors. The i-vector extractor principle is the same, i.e. it explores how an utterance specific model differs from a UBM due to the unique characteristics of the utterance. However, the

UBM is not a GMM anymore. Now this role is played by a DNN, discriminatively trained to discern phoneme senones. This neural network is now in charge of the responsibilities γ_{nc} required to compute the Baum Welch statistics N_c and $\overline{\mathbf{F}}_c$, the unique input for i-vector training. However, due to the fact that no GMM-UBM is involved, γ_{nc} now represents the probability of the feature frame o_n to contain the the senon c instead.

An alternative proposal are phonetic i-vectors [Viñals et al., 2019d]. This proposal sets an original i-vector model in which the GMM-UBM responsibility depends on a prior activation, controlled by a DNN phoneme classifier. Under this approach, the set of C components is decomposed in multiple subsets, each one responding to individual phonemes. By doing so, particular phoneme models become more specific while reducing acoustic uncertainties.

2.5.3.3 DNN embeddings

The improvements of hybrid i-vectors were outstanding, outperforming past technologies. The results in [Sadjadi et al., 2016] presented an unprecedent performance combining DNN posteriors with BNFs. However, technologies were still suffering from i-vectors flaws.

The proposed evolution was a cutting-edge idea. Rather than evolving the generative i-vector model, it trains a totally discriminative DNN. In [Snyder et al., 2016] x-vectors were proposed following this idea: A neural network is train to classify an audio among a closed set of speakers. The input features first undergo multiple frame-level non-linear transformations and then they are pooled into an utterance projection. This projection goes through utterance-level non-linear transformations before its classification. The network is trained to recognize a large pool of speakers by means of cross entropy. Given a trained network, the embeddings also known as x-vectors are extracted during the forward propagation of the information, in the utterance-level transformations.

The great performance of x-vectors has encouraged the community to evolve to DNNs. Now multiple alternatives to x-vectors are available, including LSTM based architectures [Wang et al., 2018], Wide Residual Network based embeddings [Villalba et al., 2019][Viñals et al., 2019d] or even expanded x-vectors [Villalba et al., 2019].

2.5.4 Probabilistic Linear Discriminant Analysis (PLDA)

PLDA is a statistical linear backend. Defined in [Prince and Elder, 2007] as a generative model, PLDA applies the subspace concept already considered in JFA, assuming the embedding $\boldsymbol{\phi}_j$ as a sum of variability terms:

$$\boldsymbol{\phi}_j = \boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \mathbf{U}\mathbf{x}_j + \epsilon_j \tag{2.20}$$

where $\mathbf{V}\mathbf{y}_i$ represents the speaker variability term and $\mathbf{U}\mathbf{x}_j$ the utterance variability counterpart. Both terms consist of low rank matrices ($\mathbf{V}$ and $\mathbf{U}$ respectively), which define subspaces for the latent variables $\mathbf{y}_i$ and $\mathbf{x}_j$ respectively. Whereas the speaker latent variable $\mathbf{y}_i$ is tied along all utterances with the same speaker, $\mathbf{x}_j$ is particular for each embedding j. We consider these latent variables, $\mathbf{y}_i$ and $\mathbf{x}_j$, to be standard normal distributed. Additionally, the model also includes an extra variability term ϵ_j to explain the residual variability in each particular embedding. ϵ_j is modeled by means of a zero-mean Gaussian distribution and diagonal covariance matrix $\mathbf{D}^{-1}$. Finally, $\boldsymbol{\mu}$ is the constant speaker independent term.

Although this model offers a closed-form solution, when firstly applied on embeddings (i-vectors at that time), its performance was not significatively better. It requires embeddings to be Gaussian in order to properly obtain its improvement, although the extracted i-vectors were far from this distribution. The most popular solution to this issue is length normalization [Garcia-Romero and Espy-Wilson, 2011]. Embeddings, before feeding the PLDA model, are forced to reassure that its Euclidean norm is equal to one. This process projects the input embeddings into a hypersphere of radius equal to 1. Before length-normalization, embeddings should be centered and whitened. By doing so, the resulting embeddings are spread along the hypersphere rather than being concentrated in restricted regions of the hypersphere, leading to more discriminative capabilities of the systems.

Multiple alternatives have appeared to the original PLDA model. The Simplified PLDA (SPLDA) fuses the channel and residual terms. Another alternative is the Discriminative PLDA [Cumani et al., 2013a], which trains the same model in a discriminative manner. An important alternative is the Heavy-Tailed PLDA (HTPLDA) [Kenny, 2010]. This model was proposed before length-normalization as a way to deal with non-Gaussian embeddings by modifying the prior distributions. However, after length-normalization its computational complexity discouraged its usage. Nevertheless, with the advent of DNN embeddings, far more non-Gaussian than i-vectors, HTPLDA provides small benefits with respect to other alternatives [Brummer et al., 2018].

2.6 Clustering

The clustering stage in a Bottom-Up diarization architecture is responsible for the gathering of the acoustic fragments in terms of their speaker. This duty can alternatively be considered as a labeling task. Being the audio of N acoustic segments represented by the set $\mathbf{\Phi} = \{\boldsymbol{\phi}_1, .., \boldsymbol{\phi}_N\}$ of speaker representations or embeddings, clustering must infer a partition, a set of labels $\Theta = \{\theta_1, .., \theta_N\}$, so that those segments from the same speaker share a common label.

Table 2.1: Bell number B in terms of the number of elements to cluster

Number of segments N	Number of partitions B
1	1
2	2
3	5
4	15
5	52
6	203
...	...
10	115975
...	...
20	51724158235372

To do so, we first require a measure to determine how a certain partition Θ fits the set of embeddings $\mathbf{\Phi}$. This metric may have multiple natures, e.g. statistical, graphs, kernels, etc. Then, given the chosen metric we must find the partition with the best metric value. Unfortunately, regardless of the metric they all share the same difficulty: The best partition is only guaranteed to be obtained if all possible partitions are analyzed, just choosing the one with the best metric. This option is usually referred as brute-force approach. Unfortunately, studies such as [Brummer and de Villiers, 2010] reveal that the number of partitions increase very fast as long as the number of segments to cluster N rises. In fact, except for very low values of N, brute-force approaches are in general not viable.

Given an audio with N segments, the total number of possible partitions is described by the Bell number B. This number, applicable for any clustering task, represents the total number of independent possible grouping arrangements, in our case partitions, for a set of N elements. This number is defined by means of a recurrent relation:

$$B_{N+1} = \sum_{n=0}^{N} \binom{N}{n} B_n \tag{2.21}$$

$$B_0 = 1 \tag{2.22}$$

This number increases very fast as long as the value of N rises. In Table 2.1 values for low values of N are shown.

According to Table 2.1, even very low values of N imply huge number of candidate partitions. If we consider even higher values of N, e.g. 100 or 200 segments for a one-hour TV show, the number of candidate hypotheses to compare becomes intractable. Fortunately, not all

possible diarization candidates given by the Bell number make sense. The Bell number includes all possible arrangements, including the most extreme partitions, i.e. the coarsest partition with a single cluster and the finest arrangement with as many clusters as segments. While the former partition is reasonable for diarization, the latter may not, specially when N increases. When audios get longer, certain speakers should gain relevance and contribute with more than a single intervention. Besides, long interventions may be split in multiple parts during segmentation and thus multiple segments are likely to share the same speaker. For both reasons we may prefer stablishing a reasonable upper bound k, restricting the maximum number of possible clusters. This scenario has been mathematically defined as well, thanks to the Stirling numbers of second kind. The Stirling numbers of second kind $S(N,k)$ represent the total number of arrangements for N elements with the restriction of k clusters. They are defined as

$$S(n,n) = 1 \tag{2.23}$$

$$S(n,k) = \frac{1}{k!}\sum_{i=0}^{k}(-1)^i\binom{k}{i}(k-i)^n \tag{2.24}$$

$$S(n,1) = 1 \tag{2.25}$$

Because the Stirling numbers represent all the combinations of N elements into k clusters, we can relate them with the Bell number. The relationship between both numbers is:

$$B_N = \sum_{k=0}^{N} S(N,k) \tag{2.26}$$

This relationship is specially interesting when we can assume some restrictions to the label distribution of N elements. Thus, we can define a bounded version of the Bell Number as B_{NLU}, representing all possible partitions of N elements with a number of clusters between the lower bound L and the upper bound U. This new term is defined as:

$$B_{NLU} = \sum_{l=L}^{U} S(N,l) \tag{2.27}$$

Unfortunately, even considering a bounded number of combinations, the exhaustive comparison of all elements is totally unfeasible, in spite of the advent of more and more powerful computation capabilities. In Table 2.2 we analyze the complexity of the clustering task for two typical shows in the Broadcast environment: the weather forecast and the news. For both of them we analyze how many diarization partitions could we infer and how long it takes to evaluate them all. For this conceptual evaluation we assume average lengths for the shows and the

Table 2.2: Approximated computation time to carry out search for diarization for different types of content. Estimated audio time to diarize, the mean time per utterance and the time per evaluation. Assumed an evaluation time of 0.000001 seconds.

Audio Time	Segm. mean time	N	B_N	B_{NLU}	Total time B_{NLU}
Weather forecast					
15 min	5 seconds	180	1.05E242	5.43E123	1.72E110 years
15 min	10 seconds	90	1.41E101	6.73E60	2.13E47 years
The news					
30 min	10 seconds	180	1.05E242	3.16E240	1.00E227 years
30 min	15 seconds	120	5.12E145	5.11E145	1.62E132 years

inferred acoustic segments. In order to estimate the number of partitions, we make use of the bounded Bell number B_{NLU}. Its lower bound is equal to 1 while the lower bound is adapted to the show, being 5 in the weather forecast experiment and 40 for the news. Finally, we assume $1\mu s$ as the time for any diarization system to evaluate each partition. This estimation is reasonable for the simplest algorithms but an underestimation for more complex alternatives.

The results in Table 2.2 reveal that the imposition of restrictions may significantly simplify the brute-force search problem. The simplification is noticeable as long as the imposed boundaries become more and more severe. However, when these boundaries are not severe the benefits are residual compared to the original Bell number. This situation may happen either because a large number of speakers is expected or because there is a large uncertainty about the number of speakers. Unfortunately, however we estimated the number of hypothesis to evaluate, both figures reveal unfeasible computational time for exhaustive search problems.

Consequently, the clustering problem requires the solution of two different subtasks which must work together. On the one hand, a metric must be developed to evaluate the quality of a certain partition, considering both the segment representations (usually embeddings) as well as the speaker labels. On the other hand, search algorithms to carry out the optimization of the metric. Because exhaustive search cannot be considered, suboptimal solutions must be considered. This type of solutions imposes restrictions about how to look for the partition with the best metric. Thus, the partition with the best measurement cannot be guaranteed to be found. Because both metrics and search are interconnected, now some of the most popular approaches are presented.

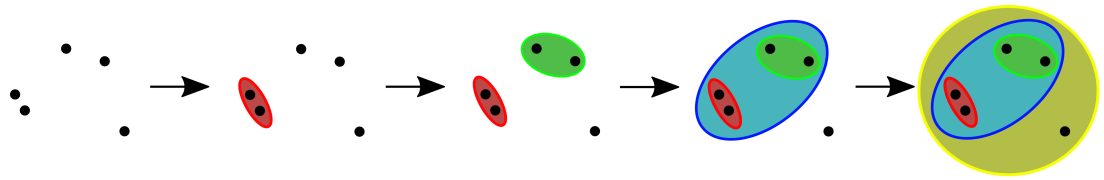


Figure 2.5: Schematic for an Agglomerative Hierarchical Clustering (AHC) performance

2.6.1 Hierarchical clustering

Hierarchical clustering is a widespread clustering solution in which we iteratively transit between the two limit partitions: All the elements belong to the same speaker (coarsest partition) and each element is responsible for an individual cluster (finest partition). Despite both directions, coarse-to-fine and fine-to-coarse, are possible and multiple contributions have been published with both of them, the latter one has gained more popularity and is usually considered when using this type of clustering for diarization. In this case we must talk about Agglomerative Hierarchical Clustering (AHC).

Although AHC was already proposed in 1970s [Duda and Hart, 1973], it was not considered for speaker clustering until [Jin et al., 1997],[Siegler et al., 1997]. Its computation requires the estimation of a pairwise affinity matrix between each pair of embeddings to cluster. Once the whole matrix is constructed, the AHC algorithm is simple. In the initial state the method considers the finest partition, i.e., as many clusters as segments. Then, iteratively those two clusters with highest affinity are merged together. Then, the affinity metric between the recently created cluster and the remaining clusters is updated. This process is repeated until a single cluster contains all elements to cluster.

The resulting structure is a decision tree describing at each iteration which clusters to merge. This tree, with as many levels as segments to cluster, contains a different number of clusters at each level. Hence by means of a stopping criterion we can infer the number of clusters in a given audio. A graphical representation is shown in Fig. 2.5.

Regarding the affinity matrix, we can make use of the same distances we previously described during segmentation. These metrics measure the benefits for two clusters to be together in comparison of being independent. Some of the already used metrics in clustering are ΔBIC [Chen and Gopalakrishnan, 1998] and KL2 [Siegler et al., 1997]. Even DNNs have been proposed for this purpose as in [Miasato Filho et al., 2018]. Furthermore in [van Leeuwen, 2010] exploited the evolutions of speaker recognition, considering the score of speaker verification systems as the metric for clustering.

With respect to the stopping criterion, the most widespread solution consists of a threshold

for the distance metric. This threshold is usually finetuned experimentally. For instance, in [Chen and Gopalakrishnan, 1998] the stopping criterion chooses the first level in which ΔBIC distance is lower than zero for all possible pair of clusters. Finally, for those cases where speaker verification score is used, rather than fixing the threshold, scores can be calibrated [Brummer and Du Preez, 2006] to properly use a known threshold.

2.6.2 Statistical approaches

The statistical approach bases the inference of the set Θ_{diar} as those labels that best explains the set of embeddings $\mathbf{\Phi}$. Mathematically, we express this condition by means of a Maximum a Posteriori (MAP) inference as:

$$\Theta_{\text{diar}} = \arg\max_{\Theta} P\left(\Theta|\mathbf{\Phi}\right) \tag{2.28}$$

By the application of Bayes rule, the set Θ_{diar} can also been interpreted as:

$$\Theta_{\text{diar}} = \arg\max_{\Theta} P\left(\mathbf{\Phi}|\Theta\right) P\left(\Theta\right) \tag{2.29}$$

In eq. 2.28 we have mathematically defined the two subtasks in clustering. First, the metric role is now played by the posterior probability $P\left(\Theta|\mathbf{\Phi}\right)$. Secondly, the best labels are those which obtain the highest probability.

The first step, the definition of the posterior distribution $P\left(\Theta|\mathbf{\Phi}\right)$, is not straightforward. In fact, this duty is almost impossible to do for an unknown audio. Therefore, we must consider eq. 2.29 and decompose the previous target distribution into the likelihood $P\left(\mathbf{\Phi}|\Theta\right)$ and the label prior $P\left(\Theta\right)$.

Regarding $P\left(\mathbf{\Phi}|\Theta\right)$ multiple options have been considered. Some options have relied on Gaussians [Gish et al., 1991] or GMMs [Ben et al., 2004]. Moreover, more elaborated models include speaker characterization concepts, especially the use of speaker latent variables. In this case, models rely on a set $\mathbf{Y}$ of I speaker latent variables $\mathbf{y}_i$, like $\mathbf{Y} = \{\mathbf{y}_1, .., \mathbf{y}_i, .., \mathbf{y}_I\}$. Each latent variable is responsible for all segments from the same speaker. Some of the most advanced statistical diarization models include this sort of approach, considering i-vectors in [Diez et al., 2018], PLDA in [Villalba and Lleida, 2014] and [Diez et al., 2019] or DNN defined distributions [Zhang et al., 2019].

With respect to the prior $P\left(\Theta\right)$, its distribution is not very important, because the posterior distribution should pay more attention to the conditional term. For this reason, the prior distribution is usually chosen to properly fit the conditional term. This choice is specially important when considering Variational Bayes (VB) solutions to find an approximation of posterior

distribution $P(\Theta|\mathbf{\Phi})$. In this case the choice of conjugate priors significantly simplifies the calculations and makes possible the obtention of closed-form solutions.

The second step is the optimization of the speaker labels given a model distribution. According to our previous explanation, the optimal speaker labels are those which satisfy eqs. (2.28) and (2.29). Ideally speaking, those labels could only be reassured by a brute force approach, checking all possible partitions Θ. However, this cannot be done, so alternative approximations have been implemented.

A first approximation prefers working with eq. (2.28). Thus, it first tries to estimate the posterior probability of the labels given the data $P(\Theta|\mathbf{\Phi})$ for its later maximization. Unfortunately, not all posterior distributions $P(\Theta|\mathbf{\Phi})$ are tractable for our maximization purposes. Consequently, we can opt for dealing with approximations $\hat{P}(\Theta|\mathbf{\Phi})$ instead. A popular alternative to obtain these approximations is Variational Bayes (VB). This technique approximates the original joint posterior distribution $P(\mathbf{Z}|\mathbf{\Phi})$, where $\mathbf{Z}$ stands for the whole set of latent variables, by a product of factors q, each one only depending on an exclusive subset from $\mathbf{Z}$. In this case our interest should be focused on the factor $q(\Theta)$, the VB approximation for $P(\Theta|\mathbf{\Phi})$. Some diarization systems whose optimization is based on this mechanism are [Valente et al., 2010] [Kenny et al., 2010] [Villalba et al., 2015] [Diez et al., 2018].

Another popular alternative works with eq. (2.29). By doing so, we work in terms of the joint distribution $P(\mathbf{\Phi}, \Theta)$ trying to maximize its value, although there is no straightforward solution. Fortunately, certain models can be easily decomposed according to the chain rule, splitting our problem into local decisions where the jth embedding $\boldsymbol{\phi}_j$ and label θ_j depend on a solution to a simplified problem. This type of problems can be represented as a decision tree, a tree with as many levels as elements to cluster, and in which any node represents an assignment decision and each edge stands for a candidate transition from one decision to another one. This idea will be treated in Chapter 6.

The main difficulty when working with trees is how to analyze the maximum number of partitions without high computational and time resources. Whenever Markov assumption can be assumed, algorithms as Viterbi [Viterbi, 1967] make possible the optimal search without carrying out a brute-force approach. This idea has been deeply exploited in speech recognition when decoding sequences of features by means of HMMs [Jelinek, 1976][Rabiner, 1989]. This type of models has been tested for diarization in [Reynolds et al., 2009]. For those cases when Markov requirements are not fulfilled the whole tree is usually unfeasible to be treated by brute force. Thus, alternatives tend to simplify the problem. First, it is usually common to treat a set of embeddings as a sequence, where any element must be ordered in a sequence, usually temporal. In this scenario successful systems have been already proposed. If a single path

along the tree is analyzed [Wang et al., 2018], real time applications can be constructed.

2.6.3 Other alternatives

The agglomerative and statistical points of view are not the only options for clustering. Along the following lines we review some of the alternative proposals for clustering

First the K-means [Lloyd, 1982] algorithm deserves a mention. This algorithm was successfully integrated in [Vaquero et al., 2013], where streams of JFA eigenvoices were clustered by means of PCA and K-means as initialization for more elaborate techniques.

Other approaches follow the spectral clustering paradigm [Ng. et al., 2002]. This alternative is a graph-based technique which relies on the eigenvalue analysis of an affinity matrix, similar to the one obtained in AHC approaches. This approach includes several contributions, such as [Ning et al., 2006] [Shum et al., 2013] [Wang et al., 2018].

Kernel based approaches were also been proposed. In [Fukunaga and Hostetler, 1975] Mean-Shift was first proposed for clustering. This algorithm models the segment distribution in terms of a kernel function, and assumes a cluster per distribution mode. Thus Mean-shift is the procedure to locate those modes and assign segments to clusters in the meanwhile. This approach, with an extended analysis in [Comaniciu and Meer, 2002], has been applied in diarization in [Senoussaoui et al., 2014][Salmun et al., 2017].

2.7 Performance metrics

Diarization is a significantly complex task, in which all of the already mentioned steps are likely to cause some sort of degradation. Therefore, any summarization of such a difficult job in just a figure is not an easy task. However, many attempts have appeared since its origins. Among all the multiple options, a metric has become the most popular diarization metric. Its name is Diarization Error Rate (DER).

DER is defined as the ratio between misclassified audio and the total amount of speech audio in a recording, a simple yet effective way to measure how well diarization systems work.

$$DER = \frac{L_{\text{Error}}}{L_{\text{total}}} \tag{2.30}$$

where L_{Error} represents the total amount of misclassified audio and L_{total} the total amount of speech to evaluate.

This misclassified audio may respond to different causes. Because these causes do not overlap each other, only four different causes are responsible for all errors:

- **MISS ERROR (MISS)**. Speech audio incorrectly labeled as non-speech. This term also measures the Voice Activity Detection (VAD) performance.
- **FALSE ALARM ERROR (FA)**. Non-speech audio in which a speaker is considered to be present. Voice Activity Detection (VAD) performance is affected by this term as well.
- **SPEAKER ERROR (SPK)**. Speech misclassified as generated by an alternative speaker.
- **OVERLAP ERROR (OV)**. Periods of time when multiple speakers are simultaneously talking. This error involves the estimation of the number of speakers (underestimation when not all speakers are detected and overestimation when non-present speakers are also labeled) as well as the misclassification of the involved speakers.

Regarding the four error terms, clearly two of them, Miss Error and False Alarm, are related with segmentation, specially the VAD step. With respect to the speaker and overlap errors, these terms mainly depend on the clustering stage. However, they are treated differently. According to its definition, the overlap error involves any inference error when multiple speakers are talking. Thus, it measures miss, false alarm and speaker errors for these periods of time. Hence overlap is the most challenging error term, with several proposed contributions about its detection (e.g. [Otterson and Ostendorf, 2007][Zelenák and Hernando, 2012]) but without any functional solution yet. Therefore, in certain evaluations this term is obviated for performance comparisons.

Due to the fact that errors are non-overlapped, DER can be decomposed on multiple terms, each one evaluating the degradation due to each type of error. The alternative definition of DER is:

$$DER = \frac{L_{\text{MISS}} + L_{\text{FA}} + L_{\text{SPK}} + L_{\text{OV}}}{L_{\text{total}}} \tag{2.31}$$

$$= E_{\text{MISS}} + E_{\text{FA}} + E_{\text{SPK}} + E_{\text{OV}} \tag{2.32}$$

where E_{MISS}, E_{FA}, E_{SPK} and E_{OV} are the DER error terms for miss, false alarm, speaker and overlap causes respectively.

Despite all its benefits regarding simplicity and decomposition of error, DER presents strong limitations. Obviating the miss error and false alarm terms, directly related with the VAD performance, no further knowledge can be inferred from DER about the speaker error. A similar score is obtained if some amount of audio is misclassified, regardless of how many speakers are affected. Besides, DER considers all the audio uniformly relevant, and errors involving

the same amount of audio are equally harmful. However, in real life neither the speakers nor their speech are equally valuable, invalidating this consideration. This is specially relevant when speakers do not contribute to the audio with the same amount of speech. For example, some errors may be irrelevant for the most talkative speakers but far more significant for those speakers contributing with much less speech. Therefore, some critics about DER metric are arising while the community is eager for finding an alternative score.

In recent times some alternative metrics have also been proposed for diarization tasks. The Mutual Information (MI) metric was defined in DIHARD 2018, a diarization evaluation in difficult conditions. The idea behind MI is measuring how much information we have about the real labels provided our hypothesed partition. The proposed metric was proposed for study and was complemented by DER, which managed the leaderboard. The metric is defined as:

$$MI = \sum_{i=1}^{R}\sum_{j=1}^{S} \frac{n_{ij}}{N} \log_2 \frac{n_{ij}N}{r_i s_j} \tag{2.33}$$

where R represents the number of clusters in the reference with r_i duration each, and S stands for the number of hypothesized clusters, each one with duration s_j. Besides, the term n_{ij} is the amount of speech assigned to the speaker i in the reference and to the cluster j in the hypothesis. Finally, N symbolizes the total amount of speech.

Another alternative is the Jaccard Error Rate (JER). This metric was proposed as alternative for DER in DIHARD 2019. The first step in the evaluation is a mapping among the R clusters in the reference and S clusters in the hypothesed partition. This mapping is carried out according to the Hungarian algorithm, so each cluster in the reference will be mapped to at most one cluster of the hypothesis and vice versa. Then for each speaker in the reference we estimate:

$$JER_{ref} = \frac{FA + MISS}{TOTAL} \tag{2.34}$$

where $TOTAL$ represents is the amount of audio present in both the reference cluster ref and the mapped counterpart. If there was no paired cluster, its value would be the total amount of speech of speaker ref. FA stands for the total amount of speech not present in the reference cluster ref but considered as part of the paired grouping. Its value is 0 if no mapping for ref was carried out. Finally, $MISS$ is the amount of speech present in speaker ref but not included in the mapped counterpart. If ref speaker has no paired cluster, its value is equal to $TOTAL$.

Having defined the individual terms JER_{ref}, the Jaccard error rate for a recording is the average of specific Jaccard error rates:

$$JER = \frac{1}{R}\sum_{ref} JER_{ref} \tag{2.35}$$

Regardless of the used metrics, they do not provide any clue about the reasons for the misclassification of audio. Hence alternative metrics should be helpful to better understand the speaker error. This error is mainly generated in the clustering block, thus clustering metrics, such as the complementary clustering and speaker impurities, well described in [van Leeuwen, 2010] are suitable for this task.

The cluster impurity represents how well the clusters from a hypothesized partition contain audio from a single speaker. Defined in terms of its cluster purity counterpart, cluster impurity is minimized as long as the obtained clusters contain audio from a single speaker. However, it is not obligatory that clusters from the same speaker share the same label, invalidating the metric for diarization.

Similarly to the cluster impurity concept, we can also define the speaker impurity. This new concept describes how well the speech from a speaker is tagged with a single label, and it is minimized as long as more and more data from one speaker only requires a single label. This metric is also invalid for diarization because multiple speakers in the same cluster do not degrade the final score.

Chapter 3

Analysis of Diarization in Broadcast Data

Diarization in broadcast is a complex task, composed of a large number of subtasks working together, as seen in Chapter 2. Whilst most of the previously described techniques work well in restricted conditions as the telephone domain, diarization in broadcast data requires many particularities to be taken into consideration. In this chapter we analyze the broadcast domain, emphasizing its particularities. For this purpose, we first introduce a reference diarization system. This system will serve to explore the wide variability along broadcast data afterwards. In this analysis we cover both quantitative and qualitative results, and study how results are affected by this uncertainty. Finally, according to the analysis and obtained results, we suggest the different lines of research, some of them treated along this thesis.

3.1 The diarization reference system

The reference system considered for this analysis is an AHC-based diarization approach, very common in the literature as baseline system. This architecture follows a Bottom-Up approach, first dividing the raw signal into segments, which are later clustered according to their speaker identity. Fig. 3.1 illustrates the basic architecture of the system.

In the following lines we explain in detail the setup for each element in our system.

- **Feature Extraction** For the audio transformation into features, we strictly consider MFCCs, standard features in the state of the art. Our MFCC setup includes a 32-band Mel filter bank, and a final coefficient reduction, only considering coefficients C1-C20. The energy information is discarded too. The inferred stream of feature vectors does not include derivatives, and undergoes a normalization for its mean and variance (CMVN).

- **Segmentation** The obtained stream of feature vectors are the input for the segmentation

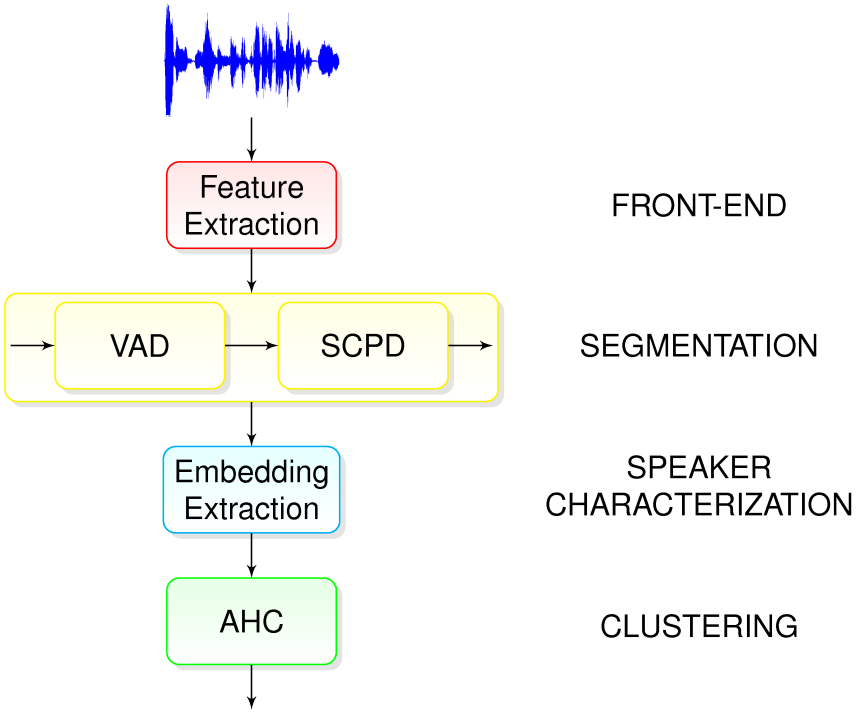


Figure 3.1: Schematic of our baseline diarization system

stage. In the reference system the segmentation step is divided into two independent subtasks, Voice Activity Detection (VAD) to differentiate speech/non-speech and Speaker Change Point Detection (SCPD) to obtain the speaker turns.

- **Voice Activity Detection (VAD)** The inference of the VAD mask is done by means of [Viñals et al., 2018a], using a segmentation-by-classification approach, which works in terms of DNNs. A 2-layer BLSTM DNN, with 256 neurons per layer, is taken into account. Each element in the second BLSTM output sequence is projected into a binary decisor. Thus, we infer one VAD label per input frame. This layer is trained and evaluated in 3-second analysis windows. Whenever the audio exceeds the window dimension, a sliding window analysis is carried out. This analysis implies a 3-second window and 2.5 seconds forward step. For the 0.5-second overlap period the inference works as follows. the first 0.25 seconds are obtained from the preceding window while the remaining 0.25 seconds are labeled with the inference from the following window. This overlapping design choice is made to avoid undesired windowing effects, specially near the artificial window borders.
- **Speaker Change Point Detection (SCPD)** For the SCPD task we rely on well known techniques. In this case we opt for a SCPD hybrid solution led by a metric-

based segmentation, in particular ΔBIC considering Gaussian distributions with full covariance matrix. We operate in a sliding window regime following the description in Section 2.4. We make use of an analysis window with a minimum length of three seconds, and a 0.25-second accumulative window expansion whenever the window does not contain any boundary. Regarding the hyperparameter λ, it is adjusted according to those results obtained during the development phase. The metric-based solution is combined with a silence-based strategy, which assumes all speech/non-speech transitions to be speaker borders. In fact, these borders are used as anchors for the Δ BIC segmentation.

- **Speaker Characterization** The estimated segments are then converted into compact representations, each one summarizing the particular feature stream for each segment. Among the multiple options described in Section 2.5, our choice for the type of representation is the i-vector. In our system i-vectors are inferred by means of an extractor of 256 Gaussians and 100-dimension total variability matrix $\mathbf{T}$. The extracted i-vectors are centered, whitened and length-normalized before feeding the clustering stage.

- **Clustering** The clustering stage in our reference system is constructed around an AHC approach, using SPLDA pairwise log-likelihood ratio (LLR) as metric. Rather than considering the original AHC that exhaustively evaluating all similarities among clusters, we follow a simplification described in [van Leeuwen, 2010]. This simplification only requests the estimation of the initial pairwise similarity among the embeddings. Then, at each fusion iteration we approximate the exhaustive similarities by approximations considering the already estimated values. Among the different options to carry out the approximation, we opt for the UPGMA (unweighted pair group method with arithmetic mean) approach [Sokal and Michener, 1958]. Regarding the clustering stop criterion, it is done by means of a threshold experimentally adjusted during development.

3.2 Analysis of broadcast data

Broadcast data is a type of domain specially characterized by its wide variability. Whenever no restrictions are applied regarding the shows of analysis, speech processing tasks must be robust enough to withstand a great range of conditions. Among the different tasks affected by this variability we must take into account diarization.

There are many reasons for broadcast data to be so mutable. From different recording locations like studio and outdoor, to the considered equipment. Furthermore, extra factors should

be considered, such as the acoustic addons, i.e. acoustic artifacts like laughter or applause, that corrupt the audio signal. For diarization purposes we will pay attention to this variability along the clustering stage. Previous blocks can be interpreted as high-quality feature extractors, thus clustering must provide the knowledge to properly group together their representations in order to obtain the final labels. During clustering we must deal with two main types of variability: the one present in the acoustic representations, the acoustic variability, and the one available in the final speaker labels, the speaker distribution variability.

The effects of both types of variability differ, specially due to their influence along the diarization pipeline. The acoustic variability is consequence of the different acoustic conditions along the different audios of interest. These different conditions cause embeddings from the same speaker to be less homogeneous, making them less robust for clustering purposes. In consequence, this variability may be responsible for a degradation of the diarization performance.

Regarding the speaker distribution variability, we are referring to the number of speakers in an audio and how much speech they contribute with. Errors in the estimation of the number of speakers are highly important because all the speech produced by the affected orators will be misclassified. Besides, the larger is the range of possible speakers of an audio, the larger are the potential errors in this estimation and more audio is usually involved. A great factor to take into account in this estimation is the distribution of speech along the different speakers. Those talkative orators with several contributions have enough audio to be robustly modeled and small misclassifications have negligible effects on them. By contrast, those speakers with very few speech are weakly represented and small errors may cause their loss.

We now present an analysis about these two types of sources of variability in broadcast data. For this purpose, we will take into consideration two large datasets: Multi-Genre Broadcast Challenge 2015[Bell et al., 2015] and Albayzín 2018 [Ortega et al., 2018]. Both datasets include a large amount of broadcast audio content covering a wide variability of shows, genres, languages and media.

3.2.1 Multi-Genre Broadcast Challenge 2015 (MGB 2015)

This dataset was released for the Multi-Genre Broadcast challenge in 2015 [Bell et al., 2015]. This challenge aims at processing tasks in the broadcast domain, including ASR, alignment and diarization. The dataset consists of approximately 1600 hours of Broadcast audio collected from British Broadcasting Corporation (BBC) along four of its channels. The total amount of audio involves around 1200 episodes from 500 different shows. All this audio is divided into three subsets: train, longitudinal development and evaluation. The subset division tries not to

share direct knowledge among the subsets, thus all episodes from a show are placed in the same subset. Aside the audio, the three subsets were distributed with diarization labels. However, the label accuracy is not uniform among subsets. Whilst the training subset includes the originally broadcast subtitles refined by a lightly-supervised ASR alignment as metadata, development and evaluation labels are manual annotations. Finally, MGB 2015 also contains an extra subset, namely development, released for the ASR evaluation. This subset consists of 28 hours from 47 shows, with manually annotated VAD marks.

3.2.2 Albayzín 2018

Albayzín 2018 is the latest edition of the Albayzín evaluations, the attempt from Red Temática de Tecnologías del Habla (RTTH) for the evolution of speech technologies in those languages spoken in the Iberian Peninsula. Regarding diarization, 2018 is the third edition after those hold in 2010 and 2016.

For the 2018 edition the evaluation consists of approximately 600 hours of data from mass media domain, covering two different languages (Spanish and Catalan) and two different mass media (TV and radio). The whole dataset was composed by three different subsets, acquired along the different editions: From 2010 edition we have available 84 labeled hours of audio from 3/24 TV channel in Catalan. These data are complemented by 2016 data: 23 hours of manually annotated audio from broadcast radio signal from Corporación Aragonesa de Radio y Televisión (CARTV) in Spanish. Finally, 2018 edition also adds around 400 hours from broadcast content from Radio Televisión Española (RTVE). The evaluation divides the pool of data as follows: For training and development both 3/24 and CARTV are available, as well as 10 hours from RTVE with manual annotations. Evaluation data consists of 40 hours from RTVE subset.

3.2.3 Acoustic variability

The richness of audio content makes both MGB 2015 and Albayzín 2018 a suitable choice to analyze variability in Broadcast data, studying how different factors affect the performance of diarization systems. For the audio variability we will make use of the diarization system paradigm, specially the SPLDA model.

According to the PLDA paradigm, SPLDA models the variability along its training data by projecting it in two subspaces, the inter-speaker space defined by the matrix $\mathbf{V}\mathbf{V}^T$ and the intra-speaker space described by matrix $\mathbf{W}^{-1}$. Moreover, due to the fact that both matrices represent covariances as well, their analysis can lead to interesting information about the variability from

Dataset	$\mathrm{tr}\left(\mathbf{V}\mathbf{V}^T\right)$ Subspace	$\mathrm{tr}\left(\mathbf{W}^{-1}\right)$ Subspace
Telephone		
SRE	0.50	0.49
Broadcast		
MGB 2015	0.19	0.86
Albayzín 2018	0.49	0.58

Table 3.1: Trace analysis for PLDA inter-speaker ($\mathbf{V}\mathbf{V}^T$) and intra-speaker ($\mathbf{W}^{-1}$) subspaces

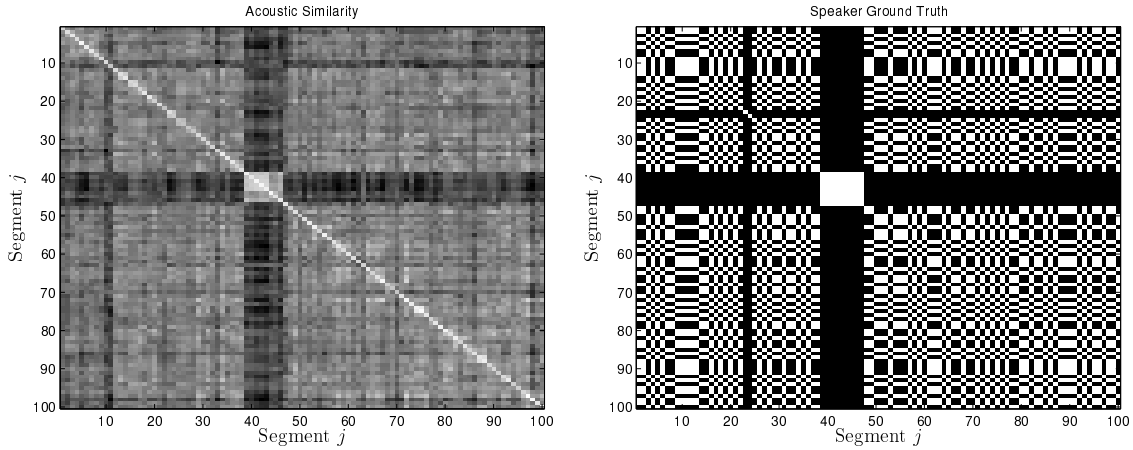
each type, inter-speaker and intra-speaker.

In Table 3.1 we carry out an analysis for both inter-speaker and intra-speaker subspaces in terms of their covariance matrices. For this analysis we study the trace of both $\mathbf{V}\mathbf{V}^T$ and $\mathbf{W}^{-1}$ matrices for our two broadcast datasets of interest, MGB 2015 and Albayzín 2018. This study approximates the total variability within each subspace, equivalent to add the variability along each dimension of the subspace as if they were independent. This study is complemented by a similar analysis for telephone channel data, which plays the role of baseline. This baseline analysis considers SRE data, constructing our models with excerpts from SRE04, SRE05, SRE06 and SRE08.

The results illustrated in Fig. 3.1 show a great mismatch in terms of conditions between telephone and broadcast data. While telephone channel presents a similar variability contained in both subspaces, our broadcast databases show at least around 18% relative extra intra-speaker variability. This measure increases up to a 352% relative extra variability in MGB dataset. Thus, when considering the broadcast domain, we must take into account the following question:

Do similar embeddings share the same speakers or just analogous acoustic conditions?

Furthermore, this intra-speaker variability is not only caused by differences among shows. In fact, broadcast data presents a high within-episode variability. This sort of variability corresponds to the different conditions in which the audio is recorded, e.g. the recording location (studio, outdoors, etc.), the involved material (microphones, postprocessing, ...), and acoustic additions (laughter, applauses, etc.). Besides, the speech signal is highly affected by the presence of emotional speech, i.e. the transformation of the voice in order to transmit extra information such as shouting (wrath), whispering (fear) or whining (pain). Regardless of the nature of the variability, it is usually very correlated along time, remaining the acoustic characteristics stable during periods of time that can contain multiple interventions from different



a) Pairwise LLR b) Ground truth mask

Figure 3.2: Section variability example. For 100 first embeddings from a Springwatch episode a) SPLDA pairwise LLR similarity metric b) Ground truth relationship.

speakers. These periods with stable conditions usually correspond to the different sections of a show. A clear example could be the news, where interventions from the news readers in studio conditions are interleaved with outdoor connections.

In order to expose this variability we again make use of our reference diarization system, studying the AHC similarity matrix constructed by means of PLDA pairwise log-likelihood ratio. This matrix should contain higher values for those elements comparing embeddings from the same speaker, regardless of the acoustic conditions. In Fig. 3.2 we illustrate the acoustic similarity matrix among the 100 first detected segments from an episode of the TV show Springwatch, from MGB 2015. In this analysis we cover an approximate 25% of the total detected interventions in the episode, balancing the tradeoff between generalization and visualization capabilities. The segments are studied in chronological order for timeline comprehension. The information includes two parts, the acoustic similarity and the ground truth mask. For the acoustic similarity we make use of the PLDA pairwise LLR, where the element ij reveals how similar are the embedding i and j. Lighter colors indicate higher speaker similarities and darker colors less probability to share the same speaker. Regarding the ground truth mask, the ij position in the figure is white if both embeddings, i and j, have the same speaker label, being black otherwise.

The two images shown in Fig. 3.2 reveal the capabilities do discern between speaker and acoustic conditions are limited. We first analyze the speaker labels of the last 50 embeddings. According to the ground truth, two speakers are responsible for a sequence of interleaved utterances, as in a dialog. However, the LLR scores did not realize about that, providing an

homogeneous score along these periods of time as if a single speaker was talking. Hence acoustic conditions must remain stable for such period of time, mitigating the speaker dissimilarity and helping for the homogeneity of the scores. By contrast, those embeddings in the range $[40, 47]$ contain an alternative speaker, clearly differentiated by LLR.

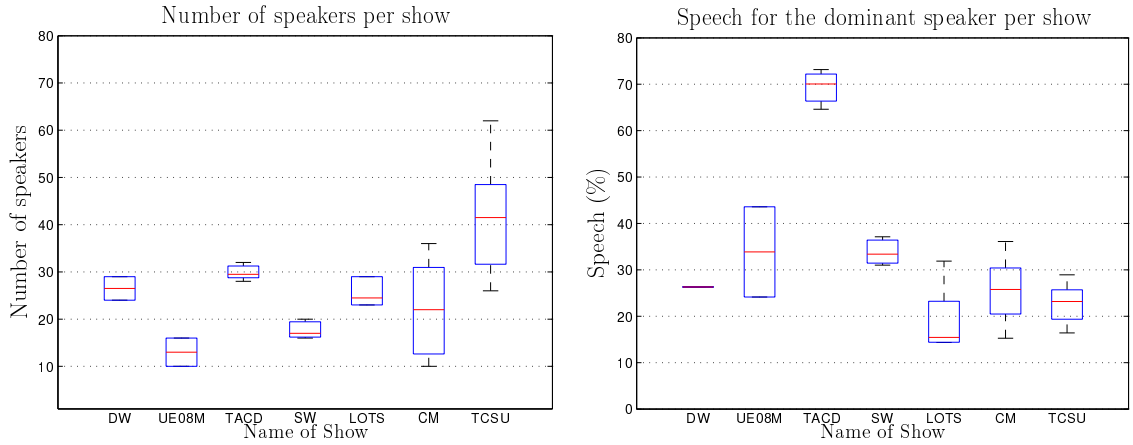
3.2.4 Variability in the speaker distribution

Another important factor for speaker diarization systems is the variability in the speaker distribution. This type of uncertainty influences on the amount of available data to characterize the speakers in an audio and also affects the stop criterion. Its influence on the stop criterion is large because this block the last step in the clustering block, choosing a partition among a range of candidates. Thus, it can be responsible for a large degradation of performance, even if the previous blocks worked perfectly. This type of variability is specially affected by the number of involved speakers and how speech is distributed among them. The wider is the range of the number of speakers the higher is the risk for large degradations. Besides, the more audio from one speaker is available, the easier to identify him or her. By contrast, the quietest speakers are in danger of being considered spurious data from a more talkative spokesperson.

We present in Fig. 3.3 an analysis of the label variability in MGB 2015. In our analysis we only study those subsets with hand-annotated data, i.e. development and evaluation. While Fig. 3.3a contains the analysis about the number of speakers per episode of each show, in Fig. 3.3b we illustrate the ratio of speech for the most talkative speaker. Each column analyzes a single show, including the interquartile range values. The involved shows in the development set are *Doctor Who* (DW), *Uefa Euro 2008 Match* (UE08M), *The Alan Clark Diaries* (TACD), *SpringWatch* (SW) and *Last of the Summer* (LOTS). With respect to the test subset, the two involved shows are *Celebrity Masterchef* (CM) and *The Culture Show Uncut* (TCSU).

According to Fig. 3.3b, the number of speakers presents significant differences among the different shows (up to 30 speakers between median values of shows 2 and 7) and within the shows, with deviations up to 20 speakers between chapters of the same show. Moreover, Fig. 3.3b, illustrates a large uncertainty as well, with differences up to 60% (*The Alan Clark Diaries* (TACD) and *Last of the Summer* (LOTS)), and presenting deviations of 10% from the median. Besides, no correlation can be observed between the variability caused by the number of speakers and the speech distribution variability.

We also have performed the same analysis on Albayzín 2018. With this subset we have just restricted the analysis to RTVE development and test audios, because are the ones that differentiate among shows. The obtained results are illustrated in Fig. 3.4. The involved shows



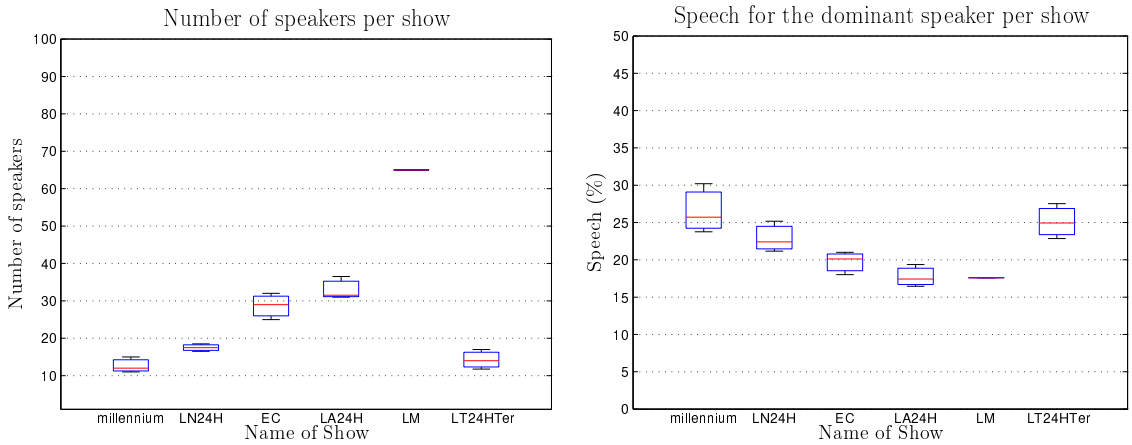
a) Number of speakers b) Proportion of speakers

Figure 3.3: Variability in the speaker distribution for MGB 2015. a) Number of speakers per show. b) Proportion of speech for the most active speaker per show. The five first shows correspond to development set, while the last two shows belong to evaluation set. For each show we represent the first, second and third quartile estimations.

for development are *millenium* and *La Noche en 24 Horas* (LN24H). Regarding the test subset, the shows *España en Comunidad* (EC), *Latinoamérica en 24 Horas* (LA24H), *La Mañana* (LM) and *La Tarde en 24 Horas Tertulia* (LT24HTer) are included.

In Fig. 3.4a we can observe that certain shows can contain up to five times as many speakers as others (*millenium* and *La Mañana*). However, Albayzín 2018 shows contain a very stable number of speakers among its episodes, in contrast to MGB 2015. Most of the shows only suffer from small variations among episodes from the same show (± 5 speakers around the median). This low intra-show but high inter-show variability encourages a specific tuning for each show, not being compatible among them. Furthermore, we have also analyzed the contribution of the most talkative speaker in Albayzín 2018. These results are shown in Fig. 3.4b. In this context Albayzín dataset behaves differently compared to MGB 2015, with no show overcoming a dominant speaker with more than 30% of the generated speech.

While the speaker predominance figure is relevant, its information is not complete. In order to obtain full knowledge we should study the whole distribution of speakers along an episode of a show. For this reason, we expand the previous analysis to single episodes. In Fig. 3.5 we illustrate the ratio of speech activity per speaker for two episodes from MGB 2015 with a very different speaker distribution. While Fig. 3.5a depicts an episode with a dominant speaker (almost 70% of the speech), Fig. 3.5b reflects a more evenly distributed speech, in which no speaker exceeds a 20% of total speech and 10 speakers make significant contributions (more than 5% of speech). The implications for this diverse distribution are relevant. For example,



a) Number of speakers b) Proportion of speakers

Figure 3.4: Variability in the speaker distribution for Albayzín 2018. a) Number of speakers per show. b) Proportion of speech for the most active speaker per show. The first two shows correspond to development set while the last four shows belong to evaluation set. For each show we represent the first, second and third quartile estimations.

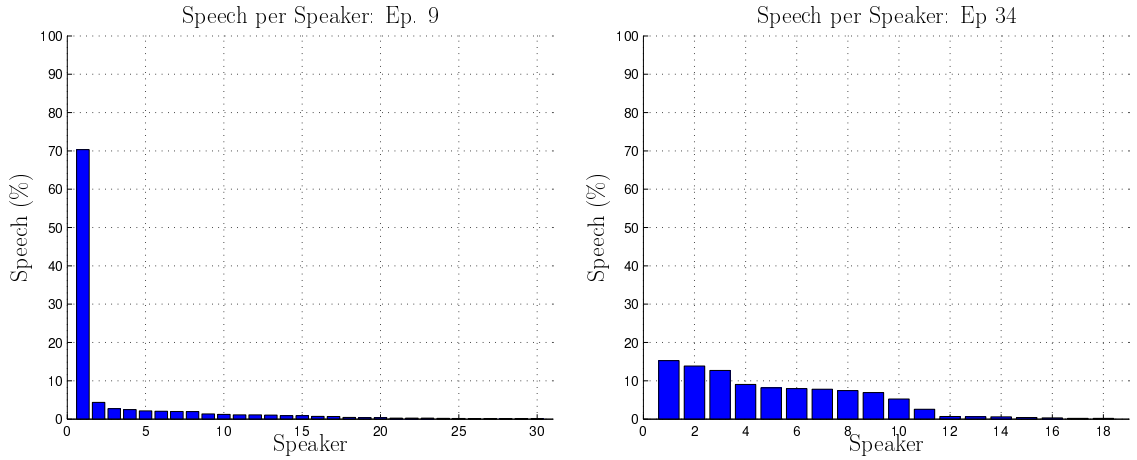
we will analyze the case when we assume a single speaker in both audios. Whilst the audio in Fig. 3.5a reassures a maximum error lower to 30% DER the show in Fig. 3.5b, under the same circumstances, should obtain approximately 90% DER.

3.3 Evaluation of performance of the diarization reference system

In the previous lines we described some of the arising difficulties of diarization when applied in the broadcast domain. However, we must also evaluate how these variations influence the performance. In the following lines we will evaluate the baseline diarization system described in Section 3.1 with our two broadcast datasets, MGB 2015 and Albayzín 2018, analyzing how variability influences the performance.

3.3.1 Evaluation of performance in MGB 2015

The first subset to evaluate with the baseline diarization system is MGB 2015. The summary with the obtained results is illustrated in Table 3.2. The results evaluate both development and test sets, for our AHC clustering (AHCPLDA). As a result guidance, Table 3.2 also includes the results for the three best systems in the original evaluation. The considered PLDA consists of a 50-dimension SPLDA. This model, trained with the available manually annotated data, receives



a) Dominant Speaker Episode b) Non-dominant Speaker Episode

Figure 3.5: Distribution of speech per speaker for two episodes: a) An episode with a dominant speaker. b) An episode with a more even speech distribution.

as input 100-dimension i-vectors, centered, whitened by means of PCA and length-normalized.

Table 3.2: DER (%) results for MGB 2015 with baseline diarization system. Results obtained for both development and test subsets. Also appended the three best systems in the original evaluation

EXPERIMENT	DEV. SET	EVAL. SET
AHCPLDA	33.47	49.39
MGB 2015 Results		
Cambridge [Karanasou et al., 2016]	N/A	40.2
ViVoLAB [Villalba et al., 2015]	N/A	43.0
LIUM [Bell et al., 2015]	N/A	44.7

Table 3.2 shows the poor performance of the agglomerative clustering in our baseline system, as well as those originally submitted to the evaluation. All systems, our baseline system as well as the evaluation ones, work around an AHC strategy. Differences among systems appear in the segmentation stage (DNN models for Cambridge and ASR tools for LIUM) as well as the addition of a resegmentation block for all evaluation systems. Thus, our baseline results are clearly worse than those already proposed in the original evaluation despite the fact that errors for all systems are in the same order. Moving to a more detailed analysis with our own results, not all the shows behave similarly. In Table 3.3 we decompose the average DER in terms of the evaluated show. The involved shows in the development set are *Doctor Who* (DW), *Uefa Euro 2008 Match* (UE08M), *The Alan Clark Diaries* (TACD), *SpringWatch* (SW) and *Last*

of the Summer (LOTS). With respect to the test subset, the two involved shows are *Celebrity Masterchef* (CM) and *The Culture Show Uncut* (TCSU).

Table 3.3: DER (%) results for MGB 2015 with baseline diarization system per show. Results shown for both development and test subsets. Avg. result stands for the overall DER value in the subset.

Dataset	Show	AHCPLDA
DEV.	DW	64.23
	UE08M	28.53
	TACD	31.04
	SW	11.60
	LOTS	51.80
	Avg.	33.47
EVAL.	CM	52.27
	TCSU	44.70
	Avg.	49.39

Results in Table 3.3 illustrate the high variability of results among the shows. Some shows are very well diarized (*Springwatch*), while others, using the same setup are highly degraded (*Doctor Who*), being up to 5 times worse diarized. Regarding those shows in the evaluation subset, both of them are highly degraded, being more noticeable for *Celebrity Masterchef*.

3.3.2 Evaluation of performance in Albayzín 2018

As well as in MGB 2015, we are interested in the performance analysis of our baseline system in Albayzín 2018. Besides, we want to study whether the previously exposed variability has its impact in performance in our results with this dataset. For Albayzín 2018 experiments the back-end is now a 100-dimension SPLDA model. This model was trained with the available training corpus, considering i-vectors centered, whitened by means of PCA and length-normalized.

Thus, in Table 3.4 we present the results obtained with our reference AHC diarization system. Similarly to those results given for MGB 2015, we also have appended the three best results provided in the original Albayzín 2018 evaluation. These three systems follow the embedding PLDA paradigm, differing on the type of considered embedding as well as different clustering strategies, some of them including a resegmentation block for label refinement.

Results in Table 3.4 show a better performance of our system with respect to MGB 2015 dataset. Moreover, with this data our AHC architecture has a similar performance to one of the best proposed systems. By contrast, the two best published results are significantly better than

Table 3.4: DER (%) results for Albayzín 2018 with baseline diarization system. Results shown for both development and test subsets. Included results for the three best systems in the original evaluation. Those systems including * follow the rules of closed-condition of the original evaluation, only training with Albayzín corpus.

EXPERIMENT	RTVE DEV	RTVE EVAL
*AHCPLDA	18.88	26.36
Albayzín 2018 Results		
GTM-UVIGO [Lleida et al., 2019]	N/A	11.4
*ViVoLAB [Lleida et al., 2019]	N/A	17.3
ODESSA [Lleida et al., 2019]	N/A	25.9

our reference system. These benefits are consequence of evolutions in the embedding extraction stage as well as more elaborated clustering strategies, not included in our baseline system.

We can also explore how the different shows perform individually. In Table 3.5 we illustrate the DER score per show for both development and test subsets. The involved shows for development are *millenium* and *La Noche en 24 Horas* (LN24H). Regarding the test subset, the shows *España en Comunidad* (EC), *Latinoamérica en 24 Horas* (LA24H), *La Mañana* (LM) and *La Tarde en 24 Horas Tertulia* (LT24HTer) are included.

Table 3.5: DER (%) results for Albayzín 2018 with baseline diarization system per show. Results shown for both development and test subsets. Avg. result stands for the overall DER value in the subset.

Dataset	Show	AHCPLDA
DEV.	millenium	8.03
	LN24H	30.45
	Avg.	18.88
EVAL.	EC	18.13
	LA24H	16.01
	LM	37.58
	LT24HTer	36.06
	Avg.	26.36

The results contained in Table 3.5 show again some large inter-show variability, with shows 4 times more accurate than others (millenium and LM). This proportions are similar to those observed with MGB 2015, despite having scored approximately half of their error result.

3.4 Conclusions

The results obtained along the present chapter have revealed different factors for the inherent variability in broadcast data. In order to deal with the detected uncertainties, diarization systems must work in the following elements:

3.4.1 The clustering approximation

Our reference system bases its diarization choices according to an agglomerative architecture. This architecture is well-known in the community due to its simplicity. Nevertheless, more evolved solutions could obtain better diarization results. The choice of an alternative clustering procedure requires that some considerations must be taken into account.

In first place we must take care of the metric to determine the quality of the partition. Results in Fig. 3.2 illustrate the great influence of channel effects in the PLDA LLR. Improvements about the modelization of the intra-speaker variability should lead to great benefits. Moreover, we can also work in the partition measurement, combining the local information considered in AHC (pairwise similarity) with a more general point of view. Thus, alternative clustering approaches should take into consideration the implications for some of the clustering choices in the decision-making process.

Another point to focus on is the stop criterion. Many of the alternative clusterings simultaneously work with multiple partitions, which contain a wide range or speakers. Whenever comparing hypotheses, biased measurements must be compensated prior to its comparison in order to prevent significant degradations.

3.4.2 The quality of the embeddings

The observed undesired variability shown in Fig. 3.2 may not be exclusively compensated during clustering, but also during the embedding extraction. In fact, the more discriminative is the information in the embeddings, the better will be its performance during the clustering stage.

Unfortunately, embeddings include more variability that is inherent to broadcast data. We are referring to more general variability terms, such as phonetic variability and short segments. Any improvement in the management of these two variabilities would lead to a general improvement in all types of diarization, as well as in speaker recognition.

3.4.3 The domain mismatch problem

Finally, we must also cover the domain mismatch. Even if clustering systems could cover the variability in different domains, their particular characteristics should require some individual adaptation for an optimal performance.

For this reason, we can make use of the domain adaptation techniques, i.e. adapt the proposed solution to each one of the domains of interest. Another solution could be the opposite, transforming the evaluation audio to fit the training conditions. Whatever is the solution, we must face another issue: broadcast data includes several shows and genres, thus in-domain data for each of them may be limited or just unavailable.

Part II

The Clustering Problem

Chapter 4

Clustering by means of Fully Bayesian PLDA

The results obtained in Section 3.3 have shown the limitations of the baseline diarization system, specially concerning the clustering stage based on an AHC solution. Therefore, this poor performance motivates the search for alternative clustering options. One of the main drawbacks of AHC is the use of local decisions, i.e. decisions taking into account very little information, as the pairwise loglikelihood ratios between two single embeddings. Thus, we would rather prefer a clustering method whose metric evaluates the overall partition. Another request is that the optimization process simultaneously optimizes all labels. A solution fitting both requirements is the clustering by means of Fully Bayesian PLDA.

4.1 The Fully Bayesian PLDA clustering solution

This proposal of clustering was first proposed in [Villalba and Lleida, 2014] as an unsupervised clustering for model adaptation. Along the following lines we will define the model and explain how it can be used for clustering tasks, including diarization. In this process we will pay attention to its Variational Bayes (VB) decomposition, key point in this approach.

4.1.1 The Fully Bayesian PLDA (FBPLDA) model

The Fully Bayesian PLDA model [Villalba and Lleida, 2014] is a generative statistical model which describes the input embeddings in terms of latent variables, some of them tied along all embeddings from the same speaker. Based on the Simplified PLDA, the FBPLDA also describes the embedding $\boldsymbol{\phi}_j$ from the ith speaker as:

$$\boldsymbol{\phi}_j = \boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \epsilon_j \tag{4.1}$$

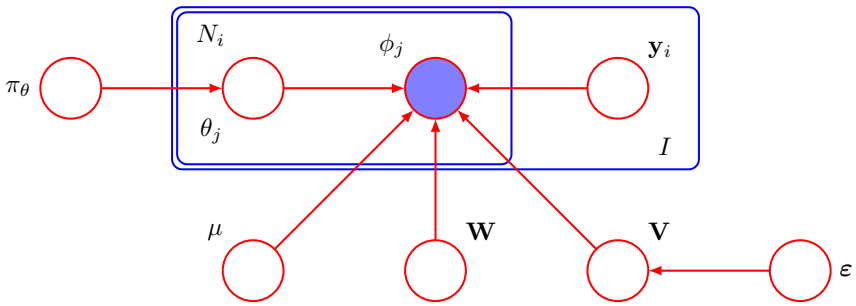


Figure 4.1: Bayesian network of the Fully Bayesian PLDA

where $\boldsymbol{\mu}$ stands for the speaker independent term. $\mathbf{V}$ represents the low-dimension matrix defining the speaker subspace. $\mathbf{y}_i$ is the speaker latent variable, standard normal distributed and common for all embeddings from the speaker i. Finally, the remaining unexplained variability in the embedding j is included by the term ϵ_j, which is modeled by means of a zero mean Gaussian with covariance $\mathbf{W}$.

The evolution of the Fully Bayesian PLDA is that, in contrast to SPLDA, speaker assignments for both training and evaluation are unknown, using latent variables instead. Thus, a set of N embeddings $\boldsymbol{\Phi} = \{\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_j, ..., \boldsymbol{\phi}_N\}$ is explained by a set of I candidate speakers, each one modeled by a speaker latent variable $\mathbf{y}_i$ from the set $\mathbf{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_i, ..., \mathbf{y}_I\}$. In order to map each embedding to its generator speaker, we consider the set of latent variables $\Theta = \{\theta_1, ..., \theta_j, ..., \theta_N\}$. Each one of the θ_j latent variables follows a multinomial distribution, which produces a one-hot sample with I values ($\theta_j = \{\theta_{1j}, ..., .\theta_{ij}, ..., \theta_{Ij}\}$). Each one of these values θ_{ij} represents the assignment of the utterance j to the ith speaker. Thus, θ_j will have its component θ_{ij} equal to one when the ith speaker is responsible for the embedding j, being zero otherwise. Taking this assignment into account, we can model $\boldsymbol{\Phi}$ in terms of $\mathbf{Y}$ and Θ as:

$$P\left(\boldsymbol{\Phi}|\mathbf{Y}, \Theta\right) = \prod_{j=1}^{N}\prod_{i=1}^{I}\mathcal{N}\left(\boldsymbol{\phi}_j|\boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i, \mathbf{W}^{-1}\right)^{\theta_{ij}} \tag{4.2}$$

Due to the Bayesian approach, the speaker labels Θ *a priori* follow a multinomial distribution. This distribution is complemented by its own prior, π_θ, which explains the multinomial weights according to Dirichlet distribution. Besides, the described Fully Bayesian PLDA proposes an extra evolution. Instead of considering point estimations for the model parameters ($\boldsymbol{\mu}$, $\mathbf{V}$ and $\mathbf{W}$), this evolution assumes them to be latent variables as well. While the mean $\boldsymbol{\mu}$ and the columns of the speaker matrix $\mathbf{V}$ are treated with a Gaussian prior, $\mathbf{W}$ is modeled in terms of a Wishart distribution. Finally, the model also includes a prior variable $\boldsymbol{\varepsilon}$ for the variable $\mathbf{V}$. The Bayesian network describing the whole model is illustrated in Fig. 4.1.

The training procedure for this model is not nearly as simple as for the SPLDA. The training of the latter model works in terms of the Expectation Maximization (EM) algorithm. This algorithm requires the estimation of the posterior distribution for each one of the latent variables in the model (E step), updating the point-estimation model parameters to maximize the loglikelihood (M step). However, in the FBPLDA model a closed-form solution for each posterior is not possible, thus E step cannot be performed. Therefore, the original work [Villalba and Lleida, 2014] also proposes an alternative training strategy by means of the Variational Bayes (VB) [Attias, 1999][Bishop, 2006].

Variational Bayes is an approximation method that allows to mimic the EM algorithm by a variational equivalent. Given a model depending on the set of latent variables $\mathbf{Z} = \{\mathbf{Z}_1, ..., \mathbf{Z}_h, ..., \mathbf{Z}_H\}$, VB approximates the posterior distribution $P(\mathbf{Z}|\mathbf{\Phi})$ by a factorial distribution $q(\mathbf{Z}) = \prod_{h=1}^{H} q(\mathbf{Z}_h)$. Each one of the obtained factors $q(\mathbf{Z}_h)$ is an approximation of the real posterior distribution $P(\mathbf{Z}_h|\mathbf{\Phi})$. In order to obtain the best approximation following the factorial restrictions each factor $q(\mathbf{Z}_i)$ must follow a distribution following the relationship:

$$\ln q(\mathbf{Z}_h) = \mathrm{E}_{\forall \mathbf{Z}_{r,r\neq h}} [\ln P(\mathbf{Z}, \mathbf{\Phi})] \tag{4.3}$$

Unfortunately, the approximation by means of a factorial distribution has limitations. Despite the fact that the obtained factor distributions $q(\mathbf{Z}_h)$ only depend on one of the latent variables $\mathbf{Z}_h$, they are not completely independent. Taking into account eq. (4.3), some dependencies remain, being each factor constructed on top of the expected values from the other factors.

A collateral effect of the Variational Bayes approximation is that the loglikelihood of the real model is no longer a suitable metric. These type of solutions works in terms of the Evidence Lower Bound (ELBO or $\mathcal{L}$).

$$\mathcal{L}(\mathbf{\Phi}) = \int q(\mathbf{Z}) \ln \left(\frac{P(\mathbf{\Phi}, \mathbf{Z})}{q(\mathbf{Z})} \right) d\mathbf{Z} \tag{4.4}$$

Both ELBO $\mathcal{L}(\mathbf{\Phi})$ and loglikelihood $\ln P(\mathbf{\Phi})$ are interconnected. In fact, the loglikelihood term is the sum of the ELBO term plus the KL divergence between the factorial distribution $q(\mathbf{Z})$ and the real posterior distribution $P(\mathbf{Z}|\mathbf{\Phi})$. We can express this as:

$$\ln P(\mathbf{\Phi}) = \mathcal{L}(\mathbf{\Phi}) + \mathrm{KL}(q(\mathbf{Z}) || P(\mathbf{Z}|\mathbf{\Phi})) \tag{4.5}$$

In fact, the ELBO and KL terms are interconnected. The maximization of ELBO makes KL divergence to be reduced, better approximating $P(\mathbf{Z}|\mathbf{\Phi})$ by means of $q(\mathbf{Z})$. As long as this approximation is more accurate, our ELBO term will be a more reliable representation of the loglikelihood from the original model.

Moving to the specific case of the Fully Bayesian PLDA, the proposed decomposition of factors is described as follows:

$$P\left(\mathbf{Y}, \Theta, \pi_\theta, \boldsymbol{\mu}, \mathbf{V}, \mathbf{W}, \boldsymbol{\varepsilon} | \boldsymbol{\Phi}\right) = q\left(\mathbf{Y}\right) q\left(\Theta\right) q\left(\pi_\Theta\right) q\left(\boldsymbol{\mu}\right) q\left(\mathbf{V}\right) q\left(\mathbf{W}\right) q\left(\boldsymbol{\varepsilon}\right) \tag{4.6}$$

Thus, for training purposes we can now perform an analogous alternative of the EM algorithm, now maximizing ELBO. The variational equivalent to the E step must iteratively update the different factors to obtain the posterior distributions, and the analogous M step will proceed to the point estimation update. This process is repeated until convergence.

4.1.2 The clustering procedure

The clustering technique by means of the FBPLDA model proposed in [Villalba and Lleida, 2014] has a statistical background. This approach assumes that diarization labels Θ_{diar} are those that best explain the embeddings $\boldsymbol{\Phi}$. Then, the way we should compare how partitions Θ explain the data $\boldsymbol{\Phi}$ is the probability $P\left(\Theta|\boldsymbol{\Phi}\right)$, as described in Section 2.6.2.

$$\Theta_{\text{diar}} = \arg\max_{\Theta} P\left(\Theta|\boldsymbol{\Phi}\right) = \arg\max_{\Theta} \int P\left(\mathbf{Z}', \Theta|\boldsymbol{\Phi}\right) d\mathbf{Z}' \tag{4.7}$$

where $\mathbf{Z}'$ represents the set of all latent variables in the model ($\mathbf{Y}$, π_θ, $\boldsymbol{\mu}$, $\mathbf{V}$, $\mathbf{W}$ and $\boldsymbol{\varepsilon}$) except for Θ.

The application of this approach to the FBPLDA model is not straightforward. The same difficulties during training are present in this approach, thus we again must rely on our VB decomposition. In consequence, we must work in terms of approximations as follows:

$$\Theta_{\text{diar}} = \arg\max_{\Theta} \int q\left(\mathbf{Y}\right) q\left(\Theta\right) q\left(\pi_\theta\right) q\left(\boldsymbol{\mu}\right) q\left(\mathbf{V}\right) q\left(\mathbf{W}\right) q\left(\boldsymbol{\varepsilon}\right) d\mathbf{Z}' = \arg\max_{\Theta} q\left(\Theta\right) \tag{4.8}$$

Despite having simplified the clustering step to the maximization of a single factor $q\left(\Theta\right)$, the same difficulties as during training remain. The proposed factors are still interconnected by means of expectations, so the optimization of the factor $q\left(\Theta\right)$ needs other factors to be optimized as well. Unfortunately, these other factors also depend on $q\left(\Theta\right)$. For this reason we work in terms of an iterative update of factors in which, starting from an initial state, we reach a maximum ELBO. This iterative process is similar to the considered EM procedure during the SPLDA training. However, in this occasion no point estimation requires update (these only affect the model parameters $\boldsymbol{\mu}$, $\mathbf{V}$, $\mathbf{W}$ and $\boldsymbol{\varepsilon}$), thus we only consider the E step.

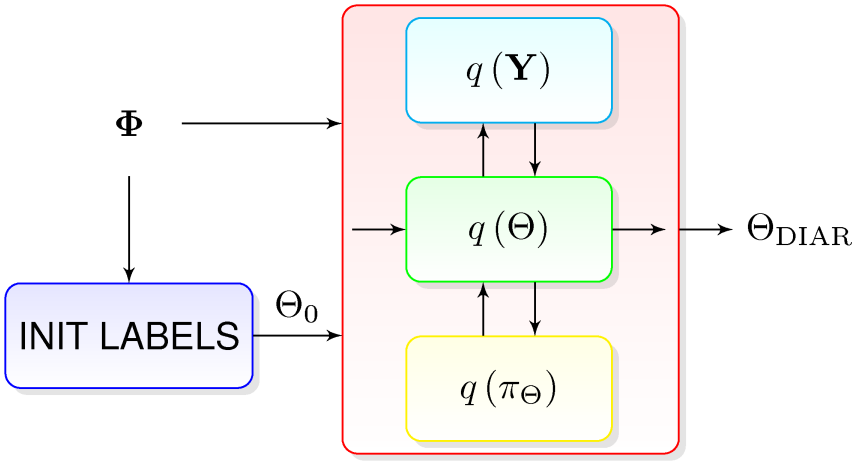


Figure 4.2: Clustering schematic based on label initialization and FBPLDA resegmentation

Moreover, because we assume the model parameters $\boldsymbol{\mu}$, $\mathbf{V}$, $\mathbf{W}$ and $\boldsymbol{\varepsilon}$ to be perfectly tuned, we exclusively reevaluate $q(\mathbf{Y})$, $q(\Theta)$ and $q(\pi_\Theta)$. This clustering procedure can be interpreted as a two-step search: The set of embeddings $\mathbf{\Phi}$ is distributed along a set of clusters $\mathbf{Y}$ during the update of the factor $q(\theta)$. Then, the same clusters $\mathbf{Y}$ are reevaluated in terms of the recently estimated Θ during the estimation of $q(\mathbf{Y})$. This iterative process may be easily understood as follows: At the beginning of each iteration, according to the current value of the speaker labels Θ we characterize each one of the considered I clusters. This characterization is done by the reevaluation of the speaker latent variables $\mathbf{Y}$ during the update of the factor $q(\mathbf{Y})$. Once the clusters are redefined, each embedding is then assigned to the most likely cluster when $q(\Theta)$ is reevaluated again.

The main disadvantage of this approach is the need for some initialization Θ_0. This initialization can be obtained in several ways, either from some prior knowledge or more often relying on the same embeddings $\mathbf{\Phi}$. Hence the proposed clustering stage follows the schematic represented in Fig. 4.2.

This clustering strategy can be interpreted as a two-step clustering: A first block is in charge of obtaining an initial partition Θ_0, which is refined afterwards by the FBPLDA clustering approach. Apart from the benefits due to label reassignment, this reclustering by means of the FBPLDA offers another advantage: an estimation about the speaker number. $q(\Theta)$ distributes the embeddings $\mathbf{\Phi}$ along I *a priori* candidate speakers. Nevertheless, it is not obligatory that all candidates generate at least one embedding. Those candidate speakers without assigned embeddings could be eliminated as part of the stop criterion.

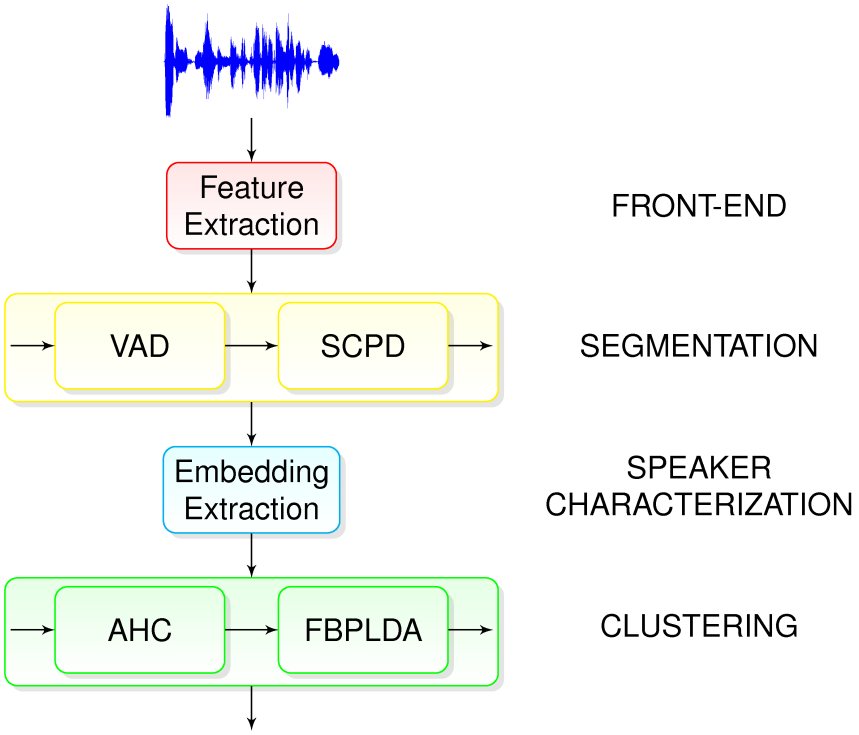


Figure 4.3: Schematic for the diarization system based on the FBPLDA resegmentation

4.1.3 Diarization using the FBPLDA model

The consideration of the FBPLDA model for diarization purposes was originally proposed in [Villalba et al., 2015]. This work proposed a diarization system whose clustering stage relied on the FBPLDA. The system is an evolution of our reference system described in Section 3.1. While the baseline system just considered a clustering block based on AHC, this new system uses the AHC stage to initialize a VB reclustering by means of the FBPLDA. The schematic for this system is illustrated in Fig. 4.3.

The performance of the new clustering stage is analyzed in Table 4.1. The analysis includes a comparison of those results obtained in Section 3.3 with our AHC reference system and the new one, with an AHC reclustered by means of the FBPLDA. The study is performed for the two broadcast datasets, MGB 2015 and Albayzín 2018, following the descriptions in Section 3.2.1 and Section 3.2.2 respectively. The analysis includes results for both development and evaluation subsets.

The obtained results evidence a consistent improvement in performance due to the FBPLDA with respect to the baseline system. Moreover, except for those results obtained with Albayzín 2018 development subset, the relative improvements overcome a 12%. This improvement is

Table 4.1: DER(%) results of the AHC and FBPLDA resegmentation based diarization systems. Included results for development and evaluation subsets from both MGB 2015 and Albayzín 2018 datasets.

EXPERIMENT	DEV. SET	EVAL. SET
MGB 2015		
AHCPLDA	33.47	49.39
AHCPLDA+FBPLDA	**23.89**	**41.58**
Albayzín 2018		
AHCPLDA	18.88	26.36
AHCPLDA+FBPLDA	**17.83**	**23.06**

specially interesting during evaluation, where we have no prior information about the evaluation conditions. Because we have seen that conditions are show dependent, an analysis per show has also been performed. In Table 4.2 we show the obtained results per show from MGB 2015. This analysis includes both development (*Doctor Who* "DW", *Uefa Euro 2008 Match* "UE08M", *The Alan Clark Diaries* "TACD", *SpringWatch* "SW" and *Last of the Summer* "LOTS") and test (*Celebrity Masterchef* "CM" and *The Culture Show Uncut* "TCSU") shows. Moreover, the shown results include both those obtained with our baseline (AHC) and those obtained with the dual clustering (AHC + FBPLDA).

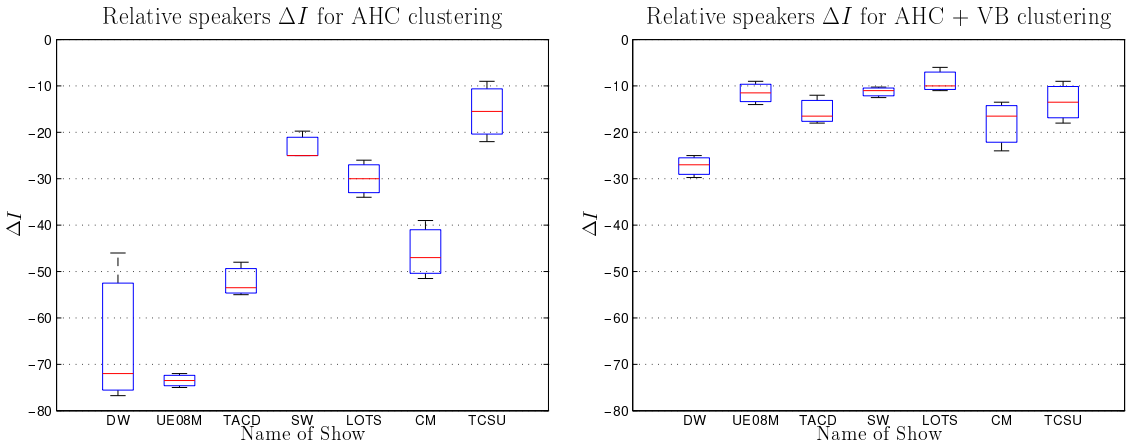
Table 4.2: DER(%) results per show of the AHC and FBPLDA resegmentation based diarization systems for MGB 2015 dataset. Avg. result stands for the overall DER value in the subset.

Dataset	Show	AHCPLDA	AHCPLDA+FBPLDA
DEV.	DW	64.23	50.07
	UE08M	28.53	12.58
	TACD	31.04	21.30
	SW	11.60	9.28
	LOTS	51.80	38.50
	Avg.	33.47	23.89
EVAL.	CM	52.27	42.04
	TCSU	44.70	40.80
	Avg.	49.39	41.58

According to the results in Table 4.2, we observe a generalized improvement for each show. This improvement is extended to all shows from both development and test subsets. Furthermore, the improvements overcome a relative 10% for all shows, and reaching 55% relative

improvement for certain shows. Being conservative, only focusing on evaluation shows, the highest improvement is approximately 20%.

Part of these benefits are due to a better estimation of the number of speakers. Whenever the FBPLDA clustering redistributes embeddings among speakers, sometimes a speaker may be left without any embedding in charge. These speakers are then discarded. In Fig. 4.4 we illustrate how the FBPLDA reclustering deals with the estimation of the number of speakers in MGB 2015 data per show. For this purpose, we represent the difference between the ground truth (or oracle) and estimated number of speakers $\Delta I = I_{ORACLE} - I_{HYP}$. This representation analyzes both the reference AHC system (Fig. 4.4a) and the FBPLDA reclustering (4.4b).



a) AHC b) AHC+FBPLDA

Figure 4.4: Analysis of $\Delta I = I_{ORACLE} - I_{HYP}$ for shows in MGB 2015 with AHC and FBPLDA resegmentation diarization systems. Results shown in boxes illustrating the 0.25, 0.50 and 0.75 percentile.

The results in Fig. 4.4 illustrate a great reduction in the estimation of the number of speakers once the FBPLDA reclustering is applied. Moreover, the same reclustering has significantly reduced the variability in this estimation. Depending on the show, the AHC system infers up to 80 extra speakers (*Doctor Who* and *Uefa Euro 2008 Match*), with intra-show variability up to 30 speakers along the episodes from the same show (Doctor who overestimates from 50 to 80 speakers depending on the episode). When the FBPLDA reclustering is applied, the value I of candidate speakers is fixed by the AHC initialization. However, the number of speakers is clearly reduced, with no show overestimating over 30 speakers, and significant reduction in the intra-show variability.

Table 4.3: DER (%) results for Albayzín 2018 depending on FBPLDA initialization. Ground truth, AHC and random initializations are considered. Both the initialization and the posterior resegmented partitions are evaluated.

Experiment	Initialization		
	Ground Truth	AHC	Random
Initialization	**4.54**	26.36	90.00
Init + FBPLDA	8.70	**23.06**	**53.08**

4.2 Analysis of FBPLDA performance

In spite of its improvements, the alternative clustering stage based on the FBPLDA is far from being perfect. Its main drawback is a consequence of the dependence with respect to the initialization Θ_0. By the iterative reevaluation of the different factors, the initial partition Θ_0 is refined so as to increase the ELBO term. However, the iterative modifications on the labels Θ only reassure to reach a local maximum. Additionally, there is no way to determine whether our solution converges to the global maximum.

Taking in mind these limitations we study the capabilities of the FBPLDA resegmentation. This analysis will pay attention to the influence of the initial labels, the FBPLDA estimation for the number of speakers and the suitability of ELBO as a quality measure. All the experiments conducted in these lines are performed with Albayzín 2018 evaluation subset.

4.2.1 Initialization impact

Our first analysis treats the relevance of the initialization for the VB solution of the FBPLDA model. In general VB solutions present a strong dependence on their initialization. If the initial partition Θ_0 is close enough to converge to the global maximum, the best possible results will be obtained. Nevertheless, in most cases this initialization is not guaranteed, hence degradations in performance appear. In Table 4.3 we present the comparison of three clustering systems working by means of an initialization block followed by the FBPLDA reclustering. The difference among systems is the initialization stage. We compare the performance when the initial partition is obtained from the reference labels (Ground Truth), obtained by our AHC solution and by random means. Due to the fact that VAD and SCPD were estimated by automatic means, they are responsible for an overall 4.54% DER degradation for the three evaluated systems.

According to the results in Table 4.3 we can see that the quality of the initialization labels is highly important for the FBPLDA reclustering performance. When a perfect initialization is provided, the reclustering stage still infers certain reassignments with respect to the initial par-

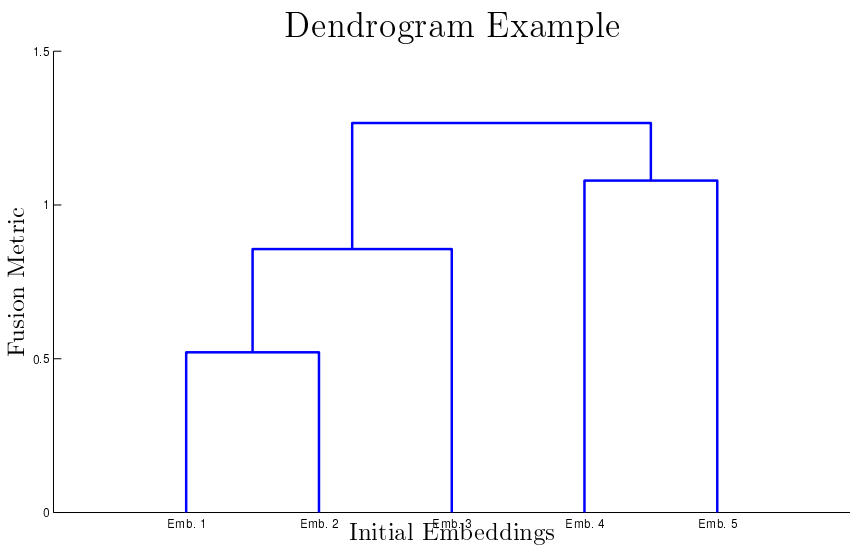


Figure 4.5: 5-level dendrogram example. Levels are shown by horizontal lines combining branches of the tree, while the last level is in the horizontal axis, where the tree leaves lie.

tition, causing small degradations. However, these changes are small enough to not completely corrupt the initialization work. Whenever imperfect initializations are provided, as our AHC option, the initial degradation is high in comparison. Thus, the FBPLDA reclustering manages to tune Θ_0 in order to obtain small improvements. However, these results are nowhere near as good as those obtained with oracle information, reflecting the importance of the global maximum. For the system with random initialization, its performance is almost negligible. However, even in this scenario the resegmentation still offers great improvements, providing much more useful labels although nowhere near as accurate as those obtained by any other initialization option.

4.2.2 Inference of the number of speakers

Due to the fact that the initial partition Θ_0 is so relevant, limiting the analysis of the FBPLDA properties to a single initialization option makes our analysis incomplete. This is why we want to test multiple real initializations and analyze the reclustering performance. For this purpose the current agglomerative clustering initialization block is very helpful. Our initialization AHC block clusters N different embeddings along N candidate partitions, each of them with a different number of speakers. These partitions can be represented in terms of a dendrogram, a decision tree with N fusions ranked in N different levels. A 5-level dendrogram example is shown in Fig. 4.5:

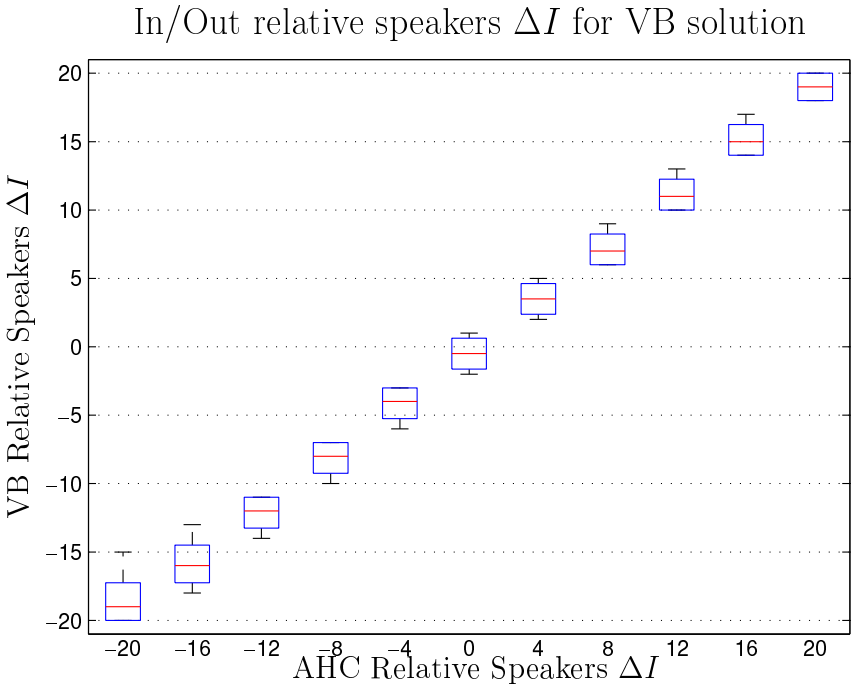


Figure 4.6: Input/output relationship for the number of speakers with FBPLDA resegmentation. Relationship expressed in terms of relative speakers ΔI. Results obtained with Albayzín 2018.

While in the AHC clustering the stop criterion works on top of this dendrogram choosing the level with the best partition, for our purposes we will get rid of it, simultaneously reclustering multiple levels of the dendrogram with the FBPLDA.

Along the following experiments we will analyze the reclustering for the different levels of the dendrogram, studying the impact of the initialization in the FBPLDA reclustering. For this purpose, each level of the AHC dendrogram is considered as an individual initial partition Θ_0, and reclustered by the FBPLDA with the same setup.

These experiments study the input/output relationship of the reclustering step in terms of the number of speakers. The experiment considers multiple levels of the AHC dendrogram, evaluating how many speakers the FBPLDA reclustering predicts. In Fig. 4.6 we represent the obtained results. Because different shows may contain a different number of speakers, the results are illustrated in terms of relative number of speakers, i.e., the difference between the number of speakers in the evaluated partition and the ground truth value ($\Delta I = I_{\text{ORACLE}} - I_{\text{ESTIM}}$). This relative number of speakers is considered for both initialization and reclustering. For this analysis we restrict the range analysis up to ± 20 relative speakers, divided into 11 even bins. For each bin we represent the first, second and third quartile estimations.

Results in Fig. 4.6 evidence an almost linear relationship between then number of the speak-

ers in the initial partition Θ_0 and those obtained after the resegmentation. Whenever Θ_0 contains less speakers than the ground truth (positive relative speakers), the FBPLDA reclustering does not discard any single speaker, only reassigning the embeddings along the different available candidate speakers. By contrast, whenever the initialization contains more speakers than the ground truth number, the algorithm starts discarding few speakers (1-3 speakers on average) although the rejection of extra speakers is not enough, significantly overestimating the number of speakers in an audio. In consequence, a bad estimation of the speaker number is difficult to be fixed by this resegmentation.

Apart from the number of speakers, diarization is affected by other factors. Another important consideration to take into account is the chance of losing real speakers. In many occasions the worthy speakers are not those who speak the most but those with few but relevant interventions. An example may be the talk shows, where the most talkative individual is the moderator despite the real valuable contributions come from the remaining speakers. Regarding the FBPLDA solution, our experimental work has revealed that it is usually reluctant to consider small sets of embeddings (sometimes a single one) as an independent speaker, opting for assuming them as spurious data from a much larger cluster. This trend of fusing small clusters with larger ones is more relevant as long as the balance of data becomes odder. By contrast, when two clusters of similar size present audio from the same speaker, the algorithm is unlikely to fuse them together.

These limitations about how the VB solution handles the resegmentation specially affects to low-talkative speakers. They contribute very little to the real audio, but depending on the application, their loss is not affordable. Therefore, we study the number of lost speakers according to the initial partition. The obtained results are shown in Fig. 4.7. Again the partitions are identified in terms of relative number of speakers ΔI. The results are also shown in terms of the first, second and third quartile.

According to Fig. 4.7 clear subclustering initializations (we assume up to 20 extra speakers) lead to the loss of 2-3 speakers on average. This trend seems steady after 12 extra speakers in our initial partition Θ_0. This result is specially interesting when compared with Fig. 4.6, which shows a growing overestimation of the speaker number, proportional to those present in the initial partition. The combination of both sources of information leads to the conclusion that we are usually losing real speakers, and most of the overestimation of speakers is a consequence of the underclustering of the remaining ones.

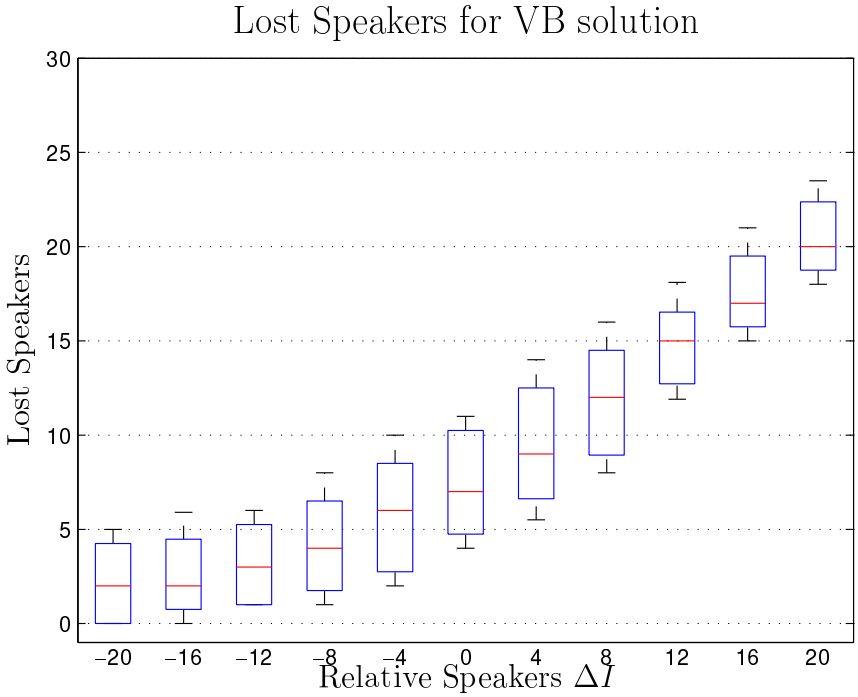


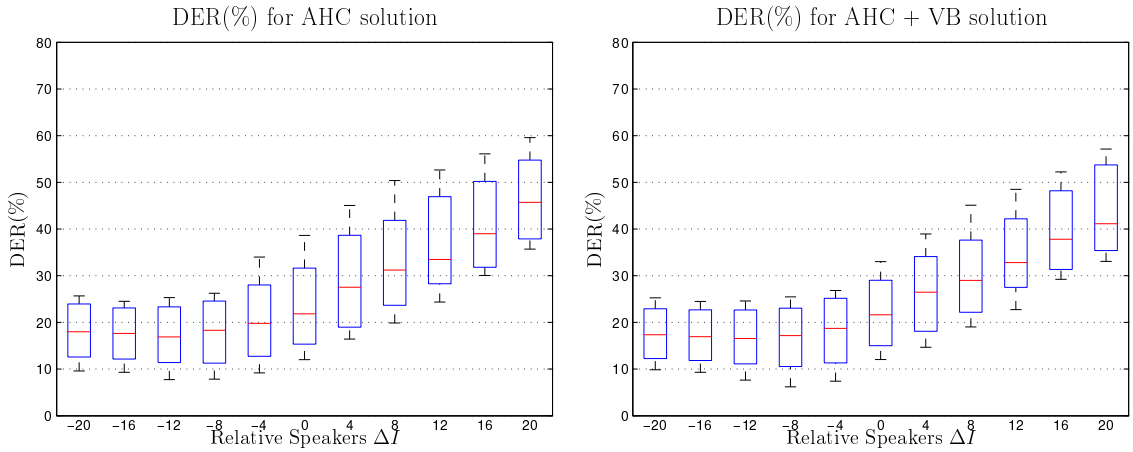
Figure 4.7: Lost speakers according to the relative number of speakers ΔI in the initial partition Θ_0. Results obtained with Albayzín 2018.

4.2.3 Number of speakers vs DER

Diarization performance does not exclusively depend on the inferred number of speakers. Actually, a good estimation about the number of speakers is not always representative for a good diarization in terms of DER. This is because DER is highly dependent on a proper identification of the largest clusters in the analysis audio. Thus, as long as really talkative speakers are properly clustered, any treatment of non-talkative speakers may be considered beneficial, even their discard.

Our next experiment studies the relationship between the initialization and the DER performance measure. For this experiment we have evaluated the inferred partitions obtained from the FBPLDA reclustering of multiple levels of the AHC dendrogram. The obtained results are illustrated in Fig. 4.8, representing the obtained DER in terms of the relative number of speakers in the initialization Θ_0. The represented information simultaneously analyzes the initialization AHC system (Fig. 4.8a) as well as the AHC block followed by the reclustering stage (Fig. 4.8b).

The results in Fig. 4.8 show that any underestimation about the number of speakers is very harmful for both diarizations, AHC and the FBPLDA reclustering. This degradation is more severe as long as the underestimation increases. These results are reasonable from the DER perspective, because severe losses of speakers will definitely cause the misclassification of very



a) AHC results b) AHC + FBPLDA results

Figure 4.8: DER (%) results for a) AHC and b) FBPLDA in terms of the relative number of speakers ΔI. Results obtained from Albayzín 2018, indicating the first, second and third quartile per bin.

talkative speakers. Interestingly, according to Fig. 4.8 when an overestimation about the number of speakers should happen to occur, the trend of DER is not so degraded, with small loses of performance in AHC and no noticeable degradation when the FBPLDA reclustering is applied. In real life applications this overestimation scenario may be worthy enough, specially considering semi-supervised applications. By means of automatic techniques diarization labels with multiple pure clusters per speaker could be easily obtained, only requiring little human supervision to match those clusters with a common speaker. This manual work is simpler than cleaning clusters with multiple speakers, i.e. the scenario in which an underestimation of the speaker number is done.

An alternative analysis is the search of the partition that provides the best diarization result. For this analysis each episode has been diarized with multiple initial partitions Θ_0, all obtained from the AHC dendrogram. The results, shown in Fig. 4.9, are compared in terms of the relative speaker of the initialization with respect to the ground truth.

According to Fig. 4.9, diarization results tend to prefer an overestimation of the number of speakers, inferring more speakers than those present in the reference labels. Moreover, the overestimation can be significative, with many episodes (more than 90% of the episodes) with at least 5 extra speakers obtaining the best DER results. These results fit with those previously obtained in Fig. 4.8, which showed that an underestimation of the speakers would lead to significant degradations.

The choice for the best initialization is a great challenge. The choice for the initial partition leading to the minimum DER may be difficult or even impossible. Even if this option was feasi-

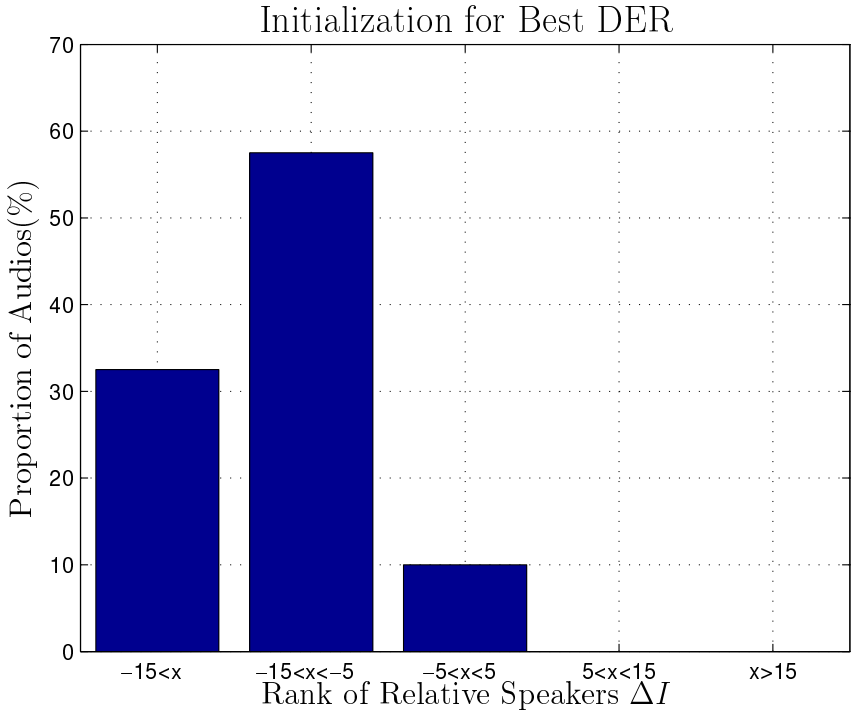


Figure 4.9: Distribution of the initialization with best DER in terms of the relative number of speakers. Results obtained from Albayzín 2018 and presented according to 5 bins.

ble, it can imply such an unaffordable computational cost. Therefore, we might sometimes seek a tradeoff, assuming certain degradations in performance if a large simplification of the systems is achieved. In Fig. 4.10 we analyze the proportion of initial partitions whose diarization output differs from the best result up to a maximum bound. This figure is composed of two different distributions, Fig. 4.10a showing the chance of a 1% DER bound and 4.10b illustrating the distribution for a 3% DER bound.

Fig. 4.10 illustrates that the probability for a partition to be under a DER degradation bound is very reduced. Only an approximate 17% of the initializations reach under the 1% DER bound, being over 33% with a higher bound (3% DER).

4.2.4 Number of speakers vs ELBO

In these lines we want to explore markers to determine whether we are working with a good partition. Even if a closed set of partitions is provided, e.g. the multiple levels of the AHC dendrogram, we need an unsupervised option to compare them and decide which one is our best option.

Because we are taking into account a statistical solution, a fair option should be the loglike-

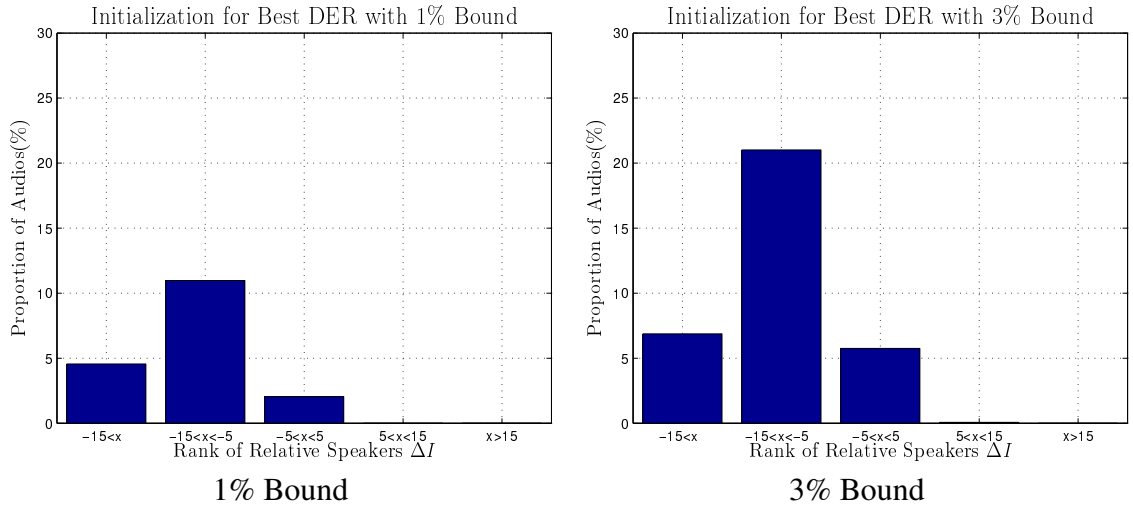


Figure 4.10: Distribution of the initialization with bounded DER, a) 1% and b) 3%, in terms of the relative number of speakers. Results obtained from Albayzín 2018 and presented according to 5 bins.

lihood. Actually, because our model is solved by means of a Variational Bayes approximation, we should consider ELBO instead. Considered as an approximation of loglikelihood, ELBO is still a good representative number about how well the partition represents the input data $\mathbf{\Phi}$.

In the next experiment we study the reliability of ELBO as partition selection criterion, exploring which initialization obtains the best ELBO after FBPLDA reclustering is performed. This result will indicate us which results are more statistically reliable. Due to range issues, we represent our results in Fig. 4.11 as a histogram illustrating the distribution of maximum ELBO in terms of the relative speaker number of the initialization Θ_0.

According to the results in Fig. 4.11, ELBO is a great indicator about the number of speakers, opting for small deviations (± 5 speakers) with respect the ground truth in almost 50% of the involved data. However, another 25% of partitions optimize ELBO by overclustering up to 15 speakers. This is specially undesirable when considering Fig. 4.10, which requests the opposite (underclustering) for a better diarization performance.

4.3 Alternative initializations

In the previous lines we have studied the great impact of the initialization on the performance in the FBPLDA reclustering solution. Its influence extends to multiple factors such as overall quality, the estimated number of speakers, how many speakers we may lose or the overall ELBO.

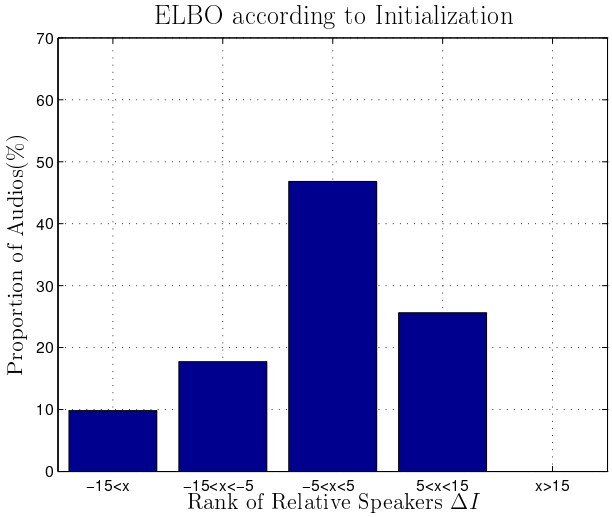


Figure 4.11: Distribution of the partition with best ELBO in terms of relative speakers. Results obtained with Albayzín 2018 and represented according to 5 bins.

All this acquired knowledge was extracted to better deal with the initialization issue. We expect to find an alternative initialization approach with respect to our first FBPLDA approach (Table 4.1), where a threshold determines the level of the dendrogram in the AHC, refined afterwards by the FBPLDA.

According to the already seen information, we illustrate two different approaches. The first one seeks an efficient tradeoff between DER improvement and computational cost. The second option tries to reach the best possible results despite falling into more elaborated strategies with a higher computational cost.

4.3.1 Computationally efficient initialization

The computationally efficient initialization approach was developed according to the information in Fig. 4.10. The illustrated information reveals that those initial partitions whose resegmentation differs from the best result below a bound are prone to contain significantly more speakers than the tuned initialization.

Our first alternative simply proposes assuming an initialization whose number of speakers is guaranteed to overcome the ground truth, thus exploiting this circumstance. By doing this, we assume an initialization in which the AHC algorithm is more unlikely to have made significant errors, and exploit the stop criteria from the FBPLDA algorithm. The great benefit of this approach is that is computationally efficient, requiring as much time as our FBPLDA baseline.

In Table 4.4 we analyze the impact of this approach considering different values for the upper number of speakers. The experiment includes both development and test subsets from

Experiment	Dev. DER(%)	Eval. DER(%)
MGB 2015		
50 Speakers	**26.08**	42.22
75 Speakers	26.13	**41.37**
100 Speakers	26.55	42.25
200 Speakers	29.68	45.93
300 Speakers	29.99	44.95
Finest partition	29.24	44.76
Albayzín 2018		
50 Speakers	**16.48**	**18.36**
100 Speakers	22.60	25.38
150 Speakers	25.94	28.14
Finest partition	60.48	62.54

Table 4.4: DER (%) results from AHC initialization with a maximum number of speakers. Included multiple maxima and the finest partition with one segment per cluster. Results shown for development and test subsets from MGB 2015 and Albayzín 2018 corpora.

both MGB 2015 and Albayzín 2018. Apart from fixed number of speakers for all episodes, our results also include the case of the finest AHC partition, i.e. one embedding per candidate speaker, as limit case.

The results in Table 4.4 show that a restricted number of speakers in both datasets may significantly improve the results, specially compared with Table 4.1. This may be a consequence of the different audio characteristics among shows, making thresholds on top of similarity metrics inappropiate. However, it is important to notice that not all initializations with a higher number are equally useful. As long as the value becomes higher, error ratios start appearing. This degradation can be taken to the limit considering the finest initialization. Therefore, this approach needs to be higher than the ground truth value and not too high for falling into degrading the performance. While in our approach we assume the same value for all episodes, their duration is not the same. Thus, more elaborated alternatives may adjust this value according to the audio duration.

4.3.2 ELBO-based initialization choice criterion

The statistical nature of the FBPLDA clustering solution makes reasonable the use of alternative initialization choices. From a statistical point of view, the best partition Θ_{DIAR} should be the one that best explains the given data $\mathbf{\Phi}$, as shown in eq. 2.28. The traditional way to measure

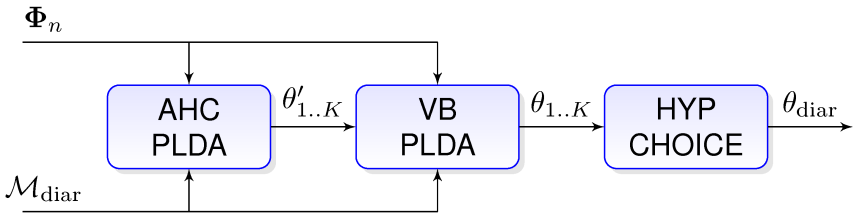


Figure 4.12: Schematic of diarization based on the simultaneous evaluation of K different initializations. The final partition is selected by means of PELBO.

how well some model represents the given data is by means of the posterior loglikelihood.

When adapting this approach to the FBPLDA model we must deal with the VB nature of our solution. This solution substitutes the original likelihood term by the ELBO term. Thus, following a similar approach our best diarization partition should be the one that maximizes the ELBO $\mathcal{L}$ term as follows:

$$\Theta_{\text{DIAR}} = \arg\max_{\Theta} \mathcal{L}(\Theta, \mathbf{\Phi}) \tag{4.9}$$

Nevertheless, this idea is not complete yet. When considering multiple initializations, not all of them initially suppose the same number of speakers. Hence the higher the number of speakers, the more likely this data can overfit to the evaluation data. Therefore, as presented in [Viñals et al., 2018a], a penalized ELBO (PELBO) term is proposed instead. This approach is inspired in BIC, in where the likelihood of the model is penalized in terms of the modelling capabilities. Then, the best labels can be obtained as:

$$\Theta_{\text{DIAR}} = \arg\max_{\Theta} \text{PELBO}(\Theta, \mathbf{\Phi}) = \arg\max_{\Theta} \left(\mathcal{L}(\Theta, \mathbf{\Phi}) - \lambda Q(\Theta)\right) \tag{4.10}$$

where $Q(\Theta)$ represents the considered excess of modeling capabilities due to the total amount of speakers in the partition Θ. This term is multiplied by a finetuning parameter λ.

This partition choice method can be applied right after the AHC algorithm, in order to choose a single partition. However, due to the great refining properties of the FBPLDA reclustering, we opted for choosing among reclustered partitions. Therefore, a set of K different initializations are simultaneously reclustered, choosing among them the final partition afterwards. This clustering structure is represented in Fig. 4.12.

In Table 4.5 we illustrate the results obtained with this algorithm. The results include DER marks for both MGB 2015 and Albayzín 2018 datasets, including both development and test subsets. Two different results are included, a single result exclusively in terms of ELBO, and the penalized ELBO as well.

Experiment	Dev. DER(%)	Eval. DER(%)
MGB 2015		
ELBO	26.82	**39.12**
PELBO	**25.95**	39.88
Albayzín 2018		
ELBO	14.48	**17.77**
PELBO	**13.90**	17.79

Table 4.5: DER (%) results for ELBO and PELBO initialization choice. Results shown for development and test subsets from both MGB 2015 and Albayzín 2018 corpora

The obtained results are significantly better than our baseline system, and also overcomes the previously described efficient solution. Moreover, both ELBO and penalized ELBO show very similar results, illustrating the robustness of the approach. Unfortunately, while penalized ELBO helps to improve simple ELBO during training, in evaluation slightly degrades in performance, partially illustrating the high domain mismatch between shows.

4.4 Conclusions

In this chapter we have explored the addition of the FBPLDA reclustering to the baseline diarization system described in Section 3.1. According to the obtained results, the performance in both broadcast diarization datasets has been significantly improved due to this block. These improvements affect both the diarization metric DER as well as the estimation of the speaker number.

Moreover, we have explored some of the FBPLDA limitations. Our analysis has revealed a great dependence of performance according to the initialization. Thus, the better the initialization, the better is the label refinement by means of FBPLDA. Besides, our study also reveals that initializations should better overestimate the number of speakers (around 10 extra speakers compared to the oracle value) so that FBPLDA provided the best labels. However, despite the improvements obtained in this overestimation scenario, we still must assume degradations as the loss of low talkative speakers. Furthermore, our analysis has determined that the Evidence Lower Bound (ELBO) seems a reasonable indicator to infer the number of real speakers within an audio.

Finally, we have explored the joint collaboration of the AHC initialization and the FBPLDA reclustering rather than assuming them as independent blocks. Thus, we proposed two success-

ful approaches where a limited number of levels of the AHC dendrogram are refined by the FBPLDA, which is also responsible for the stop criteria. Whilst our first approach explored an efficient search by assuming a single initial partition reassuring an overestimation of its speaker number, our second strategy carries out a simultaneous reclustering of multiple initializations, opting for one of the obtained partitions according to ELBO. Both of them have demonstrated that FBPLDA reclustering is a more powerful stop criteria than AHC, obtaining significant improvements. With respect to the comparison between them, the ELBO stop criteria is able to outperform the overestimation strategy, at the cost of increasing the computational costs.

Chapter 5

Uncertainty Propagation for Diarization

The great improvements obtained in Chapter 4 by means of the FBPLDA reclustering motivates the research about evolutions of this strategy. For this purpose, an alternative version of the same model is introduced in this chapter. This new proposal tries to exploit all the available information during the i-vector extraction. This extraction estimates the mean $\boldsymbol{\mu}_{\mathbf{w}_j}$ and covariance matrix $\Sigma_{\mathbf{w}_j}$ for the posterior distribution of the model latent variable $\mathbf{w}_j$ with respect to the utterance j. However, the covariance matrix, also known as uncertainty matrix, is not forwarded along traditional pipelines, only making decisions in terms of the mean, also known as i-vector.

In this chapter we present the strategy of Uncertainty Propagation (UP) and its inclusion into the i-vector PLDA framework. Later on, the properties of this inclusion are analyzed in a speaker verification task. Next, we move towards a speaker clustering evaluation stage to test this new approach. Finally, this technology is included in a new model, the Fully Bayesian PLDA with Uncertainty Propagation (FBPLDAUP), which will be tested in broadcast diarization.

5.1 Introduction

Current state-of-the-art diarization systems use embeddings as input for their clustering stage. However, these embeddings are extracted as point estimations, i.e. they map the given utterance within the embedding subspace without any confidence interval. A consequence of this consideration is that embeddings are treated as evenly robust, as well as the decisions made according to them. While in some scenarios this assumption may be valid, broadcast audio presents a large variability of conditions, such as recording equipment, location, noise, etc. Hence diarization in the broadcast domain should take into account this reliability information.

Focusing on the i-vector embedding, the procedure described in Section 2.5 explains that

during its extraction we are estimating the posterior distribution of the latent variable $\mathbf{w}_j$ given the jth utterance. By definition $\mathbf{w}_j$ follows a Gaussian distribution, depending on two parameters: its mean $\boldsymbol{\mu}_{\mathbf{w}_j}$, which maps the Gaussian within the i-vector space, and the covariance $\Sigma_{\mathbf{w}_j}$, indicating the uncertainty about the estimation. This uncertainty term is influenced by many factors, as the utterance length, acoustic conditions, or reliability of previous preprocessing. Therefore i-vectors, only considering the mean $\boldsymbol{\mu}_{\mathbf{w}_j}$ according to the literature, are losing track of useful information that cannot be recovered afterwards.

I-vector Uncertainty Propagation (UP) is the strategy in which i-vector covariances are propagated forward along the backend simultaneously with the means. This propagation seeks including some extra information into the backend so as to compensate those unreliable i-vectors and in consequence, improving the performance.

5.2 PLDA with Uncertainty Propagation (PLDAUP)

The integration of the i-vector covariance matrix (a.k.a. the uncertainty matrix) into the PLDA model is a complex process. The i-vector model defines the posterior distribution of the latent variable as a Gaussian with its own mean and covariance. From this estimation, PLDA models only take into account the mean, which is studied as a composition of factors, including the desired speaker information. In spite of this complexity, some works such as [Cumani et al., 2013b] and [Kenny et al., 2013] have successfully provided an approximation. This approach is known as PLDA with Uncertainty Propagation (PLDAUP), having several similarities with the SPLDA model. Both models represent the set $\boldsymbol{\Phi} = \{\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_j, ..., \boldsymbol{\phi}_N\}$ of N i-vectors as generated by I candidate speakers, represented by the set $\mathbf{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_i, ..., \mathbf{y}_I\}$ of speaker dependent latent variables. The assignment of each utterance to its speaker is known, being the speaker i responsible for N_i utterances. Its definition is:

$$\boldsymbol{\phi}_j \sim \mathcal{N}\left(\boldsymbol{\phi}_j | \boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \mathbf{U}_j\mathbf{x}_j, \mathbf{W}^{-1}\right) \tag{5.1}$$

where $\boldsymbol{\phi}_j$ represents the jth embedding and $\boldsymbol{\mu}$ is the constant speaker independent term. $\mathbf{V}\mathbf{y}_i$ stands for the speaker dependent term, composed by a low rank matrix $\mathbf{V}$ explaining the speaker subspace and the latent variable $\mathbf{y}_i$, tied along all utterances with the same speaker i. Moreover, the full rank matrix $\mathbf{W}$ explains the intra-speaker variability subspace. In addition to the previous terms, already present in SPLDA, an extra term $\mathbf{U}_j\mathbf{x}_j$ is added to include the i-vector covariance $\Sigma_{\mathbf{w}_j}$ information. This last term is composed of a full rank matrix $\mathbf{U}_j$, dependent on the jth utterance covariance, and a latent variable $\mathbf{x}_j$. Both latent variables, $\mathbf{y}_i$ and $\mathbf{x}_j$ have standard normal priors. The Bayesian network for the model is shown in Fig. 5.1

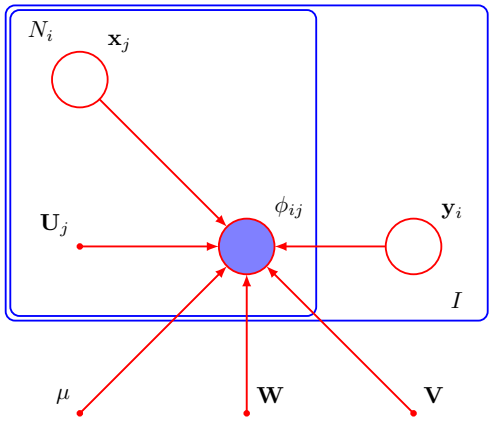


Figure 5.1: Bayesian network for PLDA with Uncertainty Propagation (PLDAUP)

The role played by $\mathbf{U}_j$ is crucial to properly include the uncertainty information into the model. Whilst previous definitions of PLDA considered all involved matrices trainable, in the PLDAUP model $\mathbf{U}_j$ is no longer an adjustable parameter. Instead, we define $\mathbf{U}_j$ in terms of $\Sigma_{\boldsymbol{\phi}_j}$, and feed it into the model as an extra input. The relationship between $\mathbf{U}_j$ and $\Sigma_{\boldsymbol{\phi}_j}$ is:

$$\mathbf{U}_j\mathbf{U}_j^T = \Sigma_{\boldsymbol{\phi}_j}^{-1} \tag{5.2}$$

The definition of $\mathbf{U}_j$ in terms of $\Sigma_{\boldsymbol{\phi}_j}$ introduces a new difficulty. Gaussian PLDAs require the input i-vectors to be length-normalized, a non-linear transformation, so as to improve their performance. However, the application of the same transformation to the matrix $\Sigma_{\mathbf{w}_j}$ is not trivial. Therefore, works as [Kenny et al., 2013] analyze multiple options trying to replicate the length-normalization in the context of matrices. One of the proposed alternatives is a scalar normalization of the uncertainty matrix, using the same factor i-vectors undergo.

$$\mathbf{L}_2 = ||\boldsymbol{\mu}_{\mathbf{w}_j}||_2 \tag{5.3}$$

$$\boldsymbol{\phi}_j = \frac{\boldsymbol{\mu}_{\mathbf{w}_j}}{\mathbf{L}_2} \tag{5.4}$$

$$\Sigma_{\boldsymbol{\phi}_j} = \frac{\Sigma_{\mathbf{w}_j}}{\mathbf{L}_2} \tag{5.5}$$

Apart from the scalar normalization, [Kenny et al., 2013] also studies the use of unscented transformations [Julier and Uhlmann, 2004] to approximate the behaviour of length-normalization in matrices. This technique proposes the sampling of the distribution $\mathbf{w}_j$ obtaining the set of 2N+1 points $a = \{a_0, ..., a_{2N}\}$, where N is the dimension of the embeddings. Next each point is transformed by the non-linear transformation $f(\cdot)$, in this case length-normalization, obtaining the set of points $b = \{b_0, ..., b_{2N}\}$ lying on the transformed space. This

latter set of point is then used to estimate the statistics of the transformed distribution. For the case of a distribution $\mathbf{w}_j$ described by its mean and variance, the unscent transformation for the function $f(\cdot)$ defines the new parameters of the new distribution ($\boldsymbol{\mu}_b$ and Σ_b) as:

$$a_s = \begin{cases} \boldsymbol{\mu}_{\mathbf{w}_j} & \text{if } s = 0 \\ \boldsymbol{\mu}_{\mathbf{w}_j} + \left(\sqrt{\frac{1}{N}\Sigma_{\mathbf{w}_j}}^{\frac{1}{2}}\right)_s & \text{if } 1 < s < N \\ \boldsymbol{\mu}_{\mathbf{w}_j} - \left(\sqrt{\frac{1}{N}\Sigma_{\mathbf{w}_j}}^{\frac{1}{2}}\right)_{s-N} & \text{if } N+1 < s < 2N \end{cases} \tag{5.6}$$

$$b = f(a) \tag{5.7}$$

$$\boldsymbol{\mu}_b = \frac{1}{2N+1}\sum_{s=0}^{2N} b_s \tag{5.8}$$

$$\Sigma_b = \frac{1}{2N}\sum_{s=0}^{2N}(b_s - \boldsymbol{\mu}_b)(b_s - \boldsymbol{\mu}_b)^T \tag{5.9}$$

where $\left(\sqrt{\frac{1}{N}\Sigma_{\mathbf{w}_j}}^{\frac{1}{2}}\right)_s$ refers to the sth column of matrix $\sqrt{\frac{1}{N}\Sigma_{\mathbf{w}_j}}^{\frac{1}{2}}$.

Before moving to the experimental work, a final consideration must be done regarding PLDAUP. The use of individual $\mathbf{U}_j$ matrices per utterance imposes some important penalties in the computational cost. The main delay is caused by the presence of these matrices in the formulation of precision matrices of the latent variables, which must be continuously reevaluated during both training and test.

5.2.1 PLDAUP in speaker recognition

To the best of our knowledge, the application of uncertainty propagation has not been evaluated in broadcast diarization. Therefore, we opt for gaining insight about this technique prior to its application in our final task. The first step is the validation of results when PLDAUP is involved. This step requires its evaluation in a speaker verification task within a restricted domain, such as telephone channel. In these experiments we just evaluate the behaviour of PLDAUP when facing short utterances.

For this experimental scenario we make use of the SRE10 "coreext-coreext det5 female" experiment. This experiment, part of the Speaker Recognition Evaluation 2010 [Greenberg et al., 2011] proposed by the National Institute of Standards and Technology (NIST), requires the scoring of more than two hundred fifty thousand trials, recorded from telephone channel in the USA. Each trial consists of two utterances, enrollment and test, with about three hundred seconds of audio per role. Each utterance is known to contain a single speaker. Evaluation rules impose no restriction about the treatment of each trial, but it is obligatory to treat trials independently, not transferring any knowledge among them.

In this scenario, the verification system follows an i-vector PLDA pipeline: Our front-end extracts a stream of MFCCs from each audio, 20 coefficients estimated for an analysis window of 25 ms with a 10 ms shift among windows. First and seconds derivatives are also calculated, and Feature Warping applied. The VAD inference is estimated by Long-Term Spectral Divergence (LTSD) [Ramirez et al., 2004]. According to these features we use a 2048-component GMM-UBM followed by a 400-dimension T-matrix. The obtained i-vectors will be projected into a 200-dimension subspace by means of LDA, and finally evaluated by means of a PLDA model of the same dimension. The baseline PLDA model will be a SPLDA version, while our new approach will evaluate the previously described PLDAUP. Neither score normalization nor calibration were applied for simplification. All stages of the system are trained with excerpts from SRE04, 05 06 and 08 [Martin and Greenberg, 2009]. In consequence, the results are measured in terms of Equal Error Rate (EER) and minDCF. This latter measure is based on the Detection Cost Function (DCF)

$$DCF = C_{\text{Miss}} P\left(\text{Miss}\right) P\left(\text{Target}\right) + C_{\text{F.A.}} P\left(\text{F.A.}\right) \left(1 - P\left(\text{Target}\right)\right) \tag{5.10}$$

This function weights the probability of missing a target ($P\left(\text{Miss}\right)$) and the probability of producing a false alarm ($P\left(\text{F.A.}\right)$ to set them in the operating point, fixed by the cost of each kind of error (C_{Miss} and $C_{\text{F.A.}}$ respectively) and the prior probability of a target trial $P\left(\text{Target}\right)$. For the evaluation, the defined operating point forces the values for the three parameters to be $C_{\text{Miss}} = 10$, $C_{\text{F.A.}} = 1$, and $P\left(\text{Target}\right) = 0.01$ respectively.

Our first experiment is designed to analyze the loss of performance when short utterances are evaluated by our standard i-vector PLDA pipeline. For benchmarking purposes we need to score the evaluation subset when long and short utterances are involved. Due to its length, approximately 5 minutes per audio, the original SRE10 coreext-coreext det5 female is a suitable experiment for long utterances. In order to carry out the same experiment with short utterances, we opted for chopping the original audios in order to fulfill the following requirements. Chops must constitute a contiguous segment within the original audio, randomly chosen in both position and length. Moreover, the chop length should contain an amount of speech within the range from 3 to 60 seconds. This chopping procedure can be interpreted as the application of a significantly more restrictive VAD mask on the original utterance. In fact, this analogy makes the comparison of experiments fair.

The evaluation of speaker verification trials requires utterances from both enrollment and test speakers. Assuming that both of them may be represented either by long or short utterances, 4 different experiments can be evaluated, named after the length of the utterances (Long and

Experiment	EER (%)	minDCF
Long-Long	3.37	0.161
Long-Short	5.98	0.291
Short-Long	5.98	0.283
Short-Short	8.76	0.403

Table 5.1: EER (%) and minDCF results with SPLDA for SRE10 coreext-coreext det5 female with involved short utterances

Short) for each role (enrollment and test). In Table 5.1 we represent the performance for each of these combinations. The performance is evaluated according to two different metrics, Equal Error Rate (EER) and minimum Detection Cost Function (minDCF).

According to the obtained results, we observe a severe degradation in performance as long as short utterances are considered. These degradations are noticeable when short utterances are involved, regardless of their role. If both roles, enrollment and test, are played by short utterances the degradation is much more significative. Furthermore, the seen degradation is noticeable in both metrics, EER and minDCF. This information is complemented by DET curves, shown in Fig. 5.2.

DET curves in Fig. 5.2 confirm those results obtained in Table 5.1, showing clear differences in performance when short utterances play any role in verification. Besides, this degradation is more noticeable when short utterances simultaneously play the enrollment and test roles. Finally, this behaviour is consistent along the whole curves, regardless of the operation point.

The previous experiment illustrates the impact of short utterances in state-of-the-art technologies and justifies the search for alternative techniques as PLDAUP. This approach is evaluated in our next experiment, scoring the same trials with long and short utterances. However, in this experiment we restrict our evaluation to two of the previous scenarios: Long-Short (original long utterance as enrollment and short utterance as test) and Short-Short (both enrollment and test are short utterances). These two conditions are evaluated with our new PLDAUP model, undergoing two different alternatives to mimic length-normalization: scalar normalization and unscent transformation. The obtained results are shown in Table 5.2, including both EER and minDCF results.

Those results in Table 5.2 illustrate the benefits due to the inclusion of Uncertainty Propagation. However, the obtained improvements do not affect the performance in the same way. While EER is clearly more affected (at least 12% relative improvements), minDCF benefits are more negligible. Besides, benefits are obtained regardless of the length-normalization approximation for matrices, although minimum extra improvements are obtained with unscent

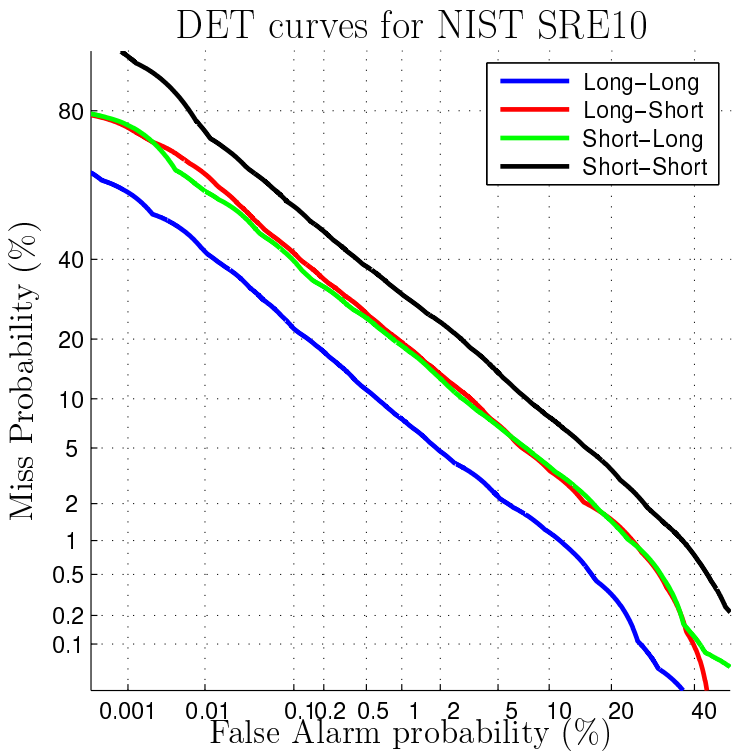
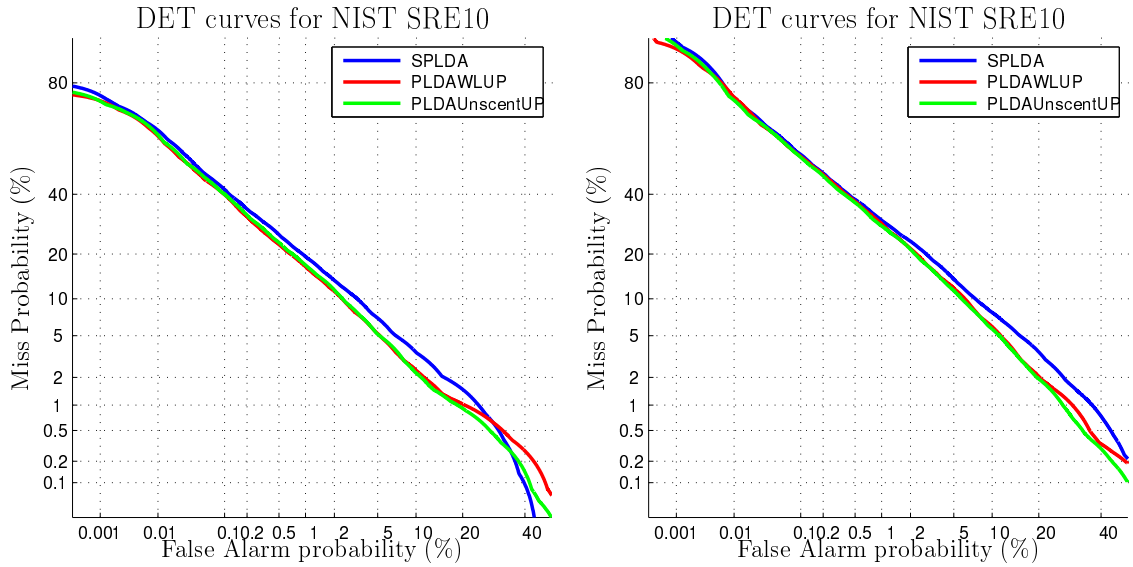


Figure 5.2: DET curves with SPLDA for SRE10 corext-coreext det5 female with involved short utterances

Experiment	Long-Short		Short-Short	
	EER (%)	minDCF	EER (%)	minDCF
SPLDA	5.98	0.291	8.76	0.403
PLDAUP scalar	5.11	**0.261**	7.72	0.389
PLDAUP unscent	**5.08**	0.269	**7.67**	**0.385**

Table 5.2: EER (%) and minDCF results with PLDAUP for SRE10 coreext-coreext det5 female with involved short utterances



a) Long-Short experiment b) Short-Short Experiment

Figure 5.3: DET curves with PLDAUP for SRE10 corext-coreext det5 female with involved short utterances

transformations. These considerations can also be observed in Fig. 5.3, where DET curves are shown.

DET curves explain the differences between minDCF and EER. PLDAUP seems to work similarly to SPLDA in those regions of the DET curve highly penalizing missing target trials. By contrast, in those regions of the curve for high false alarm is where PLDAUP obtains the highest improvements.

5.2.2 PLDAUP in speaker clustering

The inclusion of the uncertainty propagation in the PLDA for speaker verification has led to small improvements, despite not being such a revolution. However, diarization is a slightly different task. Rather than making independent decisions, diarization is the result of a large set of choices depending on each other. Thus, small individual improvements may result into an accumulation of benefits.

This is why we want to evaluate the uncertainty propagation capabilities in our clustering step, the stage where we can easily integrate our PLDAUP model. As a first approximation we do not work in diarization yet but in a speaker clustering task, working with short utterances. For this reason, we remain working in the telephone channel domain, using the same chopped subset previously used in speaker verification. The total amount of involved audios is 2740 utterances, containing 232 different speakers.

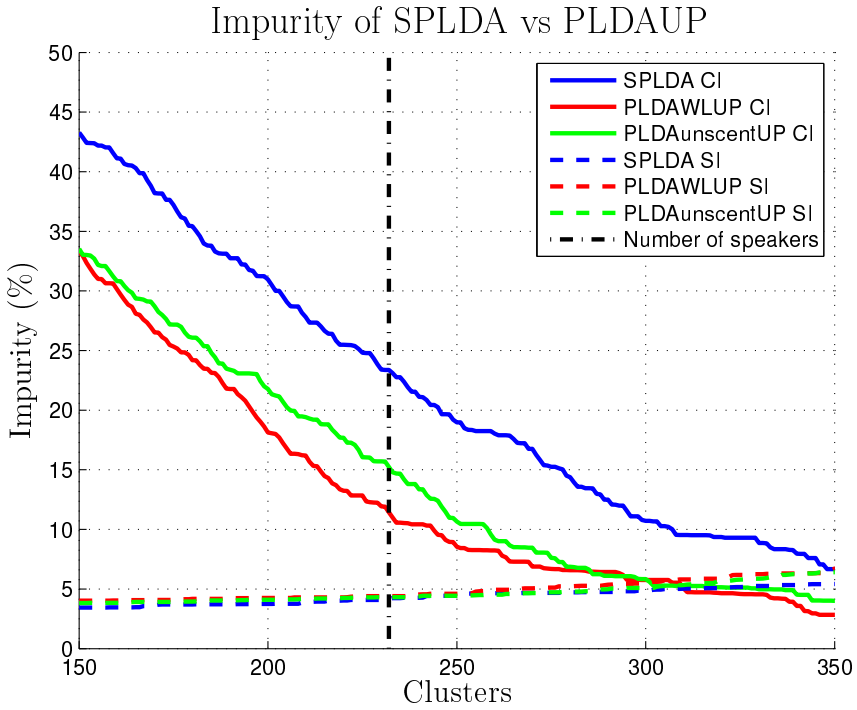
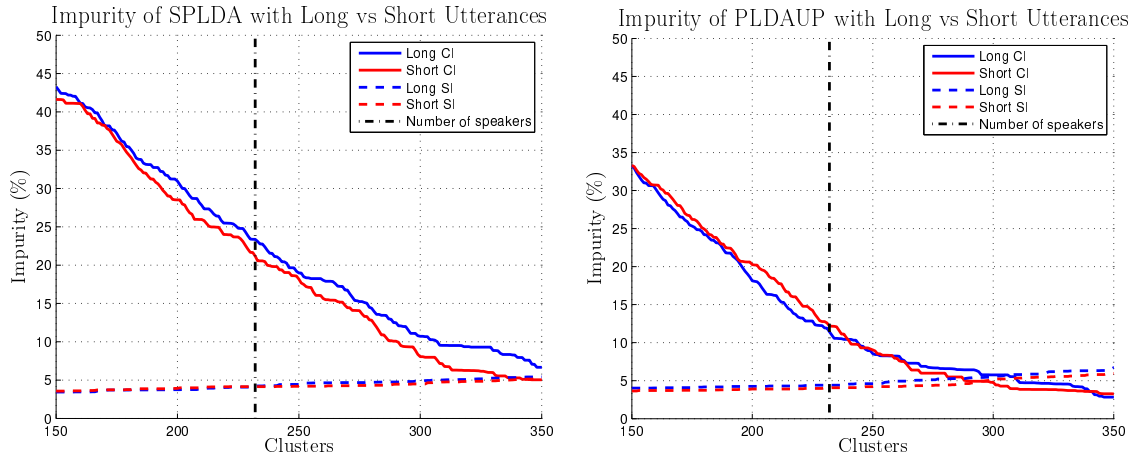


Figure 5.4: Impurity results for SPLDA and PLDAUP in SRE10 coreext-coreext det5 female chopped

Our experimental setup is the same as in our previous speaker verification experiment, i.e. a GMM-UBM i-vector extractor followed by a PLDA model. However, this time scores are not considered for 1vs1 trial decisions but the metric for an AHC solution, which determines the final labels. In order to provide a better overview about the potential of PLDAUP, we prefer not using any stop criterion, analyzing multiple levels of the AHC dendrogram. The results will be measured in terms of speaker and cluster impurities (SI and CI respectively) and shown in Fig. 5.4. Thick lines represent cluster impurities and dashed lines speaker impurities. The analysis involves 3 different PLDA versions: traditional SPLDA is shown in blue, PLDAUP with scalar normalization of the uncertainty matrix is shown in red and green represents PLDAUP with normalization of the uncertainty matrix by means of an unscent transformation. Our representation also includes an extra line (black) indicating the true value of speakers in this subset.

The results in Fig. 5.4 show a great benefit when uncertainty propagation is applied, confirming our hypothesis of improvement accumulation. For the range of study PLDAUP cluster impurity consistently undergoes an absolute improvement within the range 5-10% with respect to SPLDA, while no evident degradations in the speaker impurity are noticed. Besides, this improvement has also reduced the bias for the Equal Impurity (EI) point in terms of the number of speakers. While SPLDA reaches the EI point at 350 speakers, PLDAUP does the same at 300



a) SPLDA b) PLDAUP

Figure 5.5: Impurity results for a) SPLDA and b) PLDAUP with scalar normalization in SRE10 coreext-coreext det5 female chopped training with short utterances

speakers. Furthermore, while speaker verification results indicated that unscent transformations were better than scalar normalization of the uncertainty matrix, those obtained for the clustering task show that the scalar normalization overcomes the performance of unscent transformations up to an absolute 2-5%.

During the previous experiments we analyzed some PLDA model trained on excerpts from SRE04, 05, 06 and 08. This training pool consists of audios with large amounts of speech per utterance. Thus, training embeddings can be considered reliable enough for our standards. However, this scenario may not be so realistic in other domains, as diarization. In fact, diarization data usually consists of a combination of long and short utterances. Thus, we must analyze how our two models, SPLDA and PLDAUP, behave when short utterances are considered for model training.

For this purpose, we analyze the impact of short utterances on the training pool. In this experiment we build an alternative version of the considered training pool (SRE04, SRE05, SRE06 and SRE08) by randomly chopping the original utterances guaranteeing the speech content to be within the range of 3-60 seconds. This subset will only be considered for the training of the PLDA models, both SPLDA and PLDAUP. Under these conditions we evaluate our subset with short utterances with both PLDA models, SPLDA and PLDAUP with scalar normalization of the uncertainty matrix. In Fig. 5.5 we compare how each model responds depending on the training cohort, diving between SPLDA (Fig. 5.5a) and PLDAUP (Fig. 5.5b).

The illustrated results in Fig. 5.5 reveal interesting details. First, SPLDA seems to adapt well to short utterances, outperforming the version with long utterances for any operational

point. An explanation is that the training cohort suffers from shifts of the embeddings due to its length, similar to those in the evaluation subset. Consequently, these shifts can be considered as extra intra-speaker variability and are taken into account in the corresponding parameter ($\mathbf{W}$). By contrast, PLDAUP seems to slightly lose some of its performance. A reason for this behaviour is that intra-speaker variability lies in a subspace controlled by $\mathbf{U}_j$ and $\mathbf{W}$. When long reliable utterances are used to train the model most of this variability is forced to be in the $\mathbf{W}$ subpace, acting $\mathbf{U}_j\mathbf{U}_j^T$ as an addition during evaluation. However, when training involves short utterances both terms are representative and contributing, thus making decisions much noisier.

5.3 Fully Bayesian Probabilistic Linear Discriminant Analysis with Uncertainty Propagation (FBPLDAUP)

The confirmation of Uncertainty Propagation beneficial capabilities motivates its evaluation in broadcast diarization. However, in this domain our best results so far have been shown in Section 4.1 by means of the FBPLDA and its Variational Bayes resegmentation. This resegmentation is in fact the key point of this best approach, fixing some of the mistakes and thus improving the performance.

For this purpose, we update the FBPLDA model described in Section 4.1.1 including the new uncertainty propagation concept. The name of this new model is Fully Bayesian PLDA with Uncertainty Propagation (FBPLDAUP). This new model, as well as its predecessor, explains a set of N embeddings $\mathbf{\Phi}$ from I different speakers modeled by the set $\mathbf{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_i, ..., \mathbf{y}_I\}$, where $\mathbf{y}_i$ is a latent variable common for all utterances from the same ith speaker. Additionally, it also incorporates an extra latent variable $\mathbf{x}_{ij}$ per utterance, responsible for modeling the variability due to the utterance length. The assignment of each embedding to its responsible speaker is done in terms of θ_{ij}, a latent variable taking the value of 1 if the element j is generated by the ith speaker and 0 otherwise. Thus, we define the conditional distribution of $\mathbf{\Phi}$ as:

$$P\left(\mathbf{\Phi}|\mathbf{Y}, \Theta, \mathbf{X}, \boldsymbol{\mu}, \mathbf{V}, \mathbf{W}\right) = \prod_{i=1}^{I}\prod_{j=1}^{N}\mathcal{N}\left(\boldsymbol{\phi}_j|\boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \mathbf{U}_j\mathbf{x}_j, \mathbf{W}^{-1}\right)^{\theta_{ij}} \tag{5.11}$$

where $\mathcal{N}\left(\boldsymbol{\phi}_j|\boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \mathbf{U}_j\mathbf{x}_{ij}, \mathbf{W}^{-1}\right)$ represents the distribution of the embedding $\boldsymbol{\phi}_j$ according to speaker i. This modelization includes a speaker independent term $\boldsymbol{\mu}$, a speaker dependent term $\mathbf{V}\mathbf{y}_i$ and the i-vector variability term $\mathbf{U}_j\mathbf{x}_{ij}$. $\mathbf{V}$ is a low rank matrix explaining the speaker subspace and $\mathbf{y}_i$ is the speaker latent variable. $\mathbf{U}_j$ stands for the i-vector variability subspace

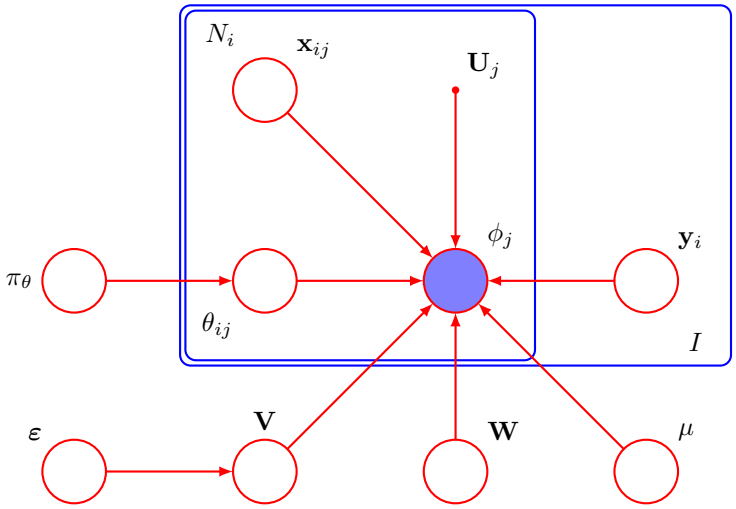


Figure 5.6: Bayesian network for the Fully Bayesian PLDA with Uncertainty Propagation

full rank matrix and $\mathbf{x}_{ij}$ its latent variable. Finally $\mathbf{W}$ is a full rank matrix explaining the within speaker subspace.

Due to the fact that we are building a Fully Bayesian solution, our model parameters ($\boldsymbol{\mu}$, $\mathbf{V}$ and $\mathbf{W}$) are distributions rather than point estimates. In fact, $\mathbf{V}$ has its own prior distribution $\boldsymbol{\varepsilon}$, a product of gamma distributions. In addition to the model parameters, the speaker labels Θ are also treated as latent variables, modeled by means of a multinomial distribution. This multinomial distribution includes a Dirichlet prior π_θ in order to explain the probabilities per class. The Bayesian network for the model is shown in Fig. 5.6.

For diarization purposes with the FBPLDAUP we follow the statistical approach already described in Section 2.6.2 maximizing the posterior distribution $P(\Theta|\mathbf{\Phi})$. However, the complexity of the model makes the true posterior intractable for optimization purposes. Hence, we prefer applying Variational Bayes for a more suitable solution. The applied simplification to the new model is:

$$P\left(\mathbf{Y},\mathbf{X},\Theta,\pi_\theta,\tilde{\mathbf{V}},\mathbf{W},\boldsymbol{\varepsilon}\right) = q\left(\mathbf{Y},\mathbf{X}\right)q\left(\Theta\right)q\left(\pi_\theta\right),q\left(\tilde{\mathbf{V}}\right)q\left(\mathbf{W}\right)q\left(\boldsymbol{\varepsilon}\right) \tag{5.12}$$

For more information about the formulation of the different priors the formulation is included in Appendix A.

Due to the relationship between FBPLDA and FBPLDAUP, both follow the diarization strategy described in Section 4.1.1, based on a Variational Bayes approximation. Our new model assumes that diarization labels Θ_{diar} should be those which best explain the given utterances,

modeled by its mean and covariance. Thus, we must maximize $P(\Theta|\mathbf{\Phi})$. By Variational Bayes we approximate this posterior distribution by $q(\Theta)$, whose maximum will be our solution. Unfortunately, VB factors are interconnected, requiring $q(\Theta)$ the other factors to be optimized as well for a proper solution, which also need $q(\Theta)$ adjusted as well. Therefore, we must apply an iterative reevaluation of factors ($q(\mathbf{Y}, \mathbf{X})$, $q(\Theta)$ and $q(\pi_\theta)$) in order to reach the best value for our hypothesis.

5.4 Diarization of broadcast data with FBPLDAUP

In the following lines we present the results in broadcast diarization when Uncertainty Propagation is included. For these experiments we make use of MGB 2015 according to the configuration explained in Section 3.2.1. The applied system follows the schematic presented in Section 4.1.2, where an AHC stage to estimate partition seeds for the VB resegmentation. In our new approach, any involved PLDA is substituted by its PLDAUP counterpart, so our AHC stage works in terms of the PLDAUP score while the VB resegmentation role is now played by the FBPLDAUP. For comparison reasons both models will present the same dimension for the speaker subspace than in the original counterpart.

In the first comparison we will compare the initialization labels. For this reason, we evaluate the whole AHC tree taking into account two similarity metrics, SPLDA and PLDAUP. Then, for each level on both trees we will evaluate the resulting partitions by means of DER and calculate $\Delta\mathrm{DER} = \mathrm{DER}_{\mathrm{PLDAUP}} - \mathrm{DER}_{\mathrm{SPLDA}}$. This comparison is repeated for each involved show in MGB 2015, including both development and test subsets. In Fig. 5.7 we illustrate a histogram about the relative variations of DER depending on the considered model.

In Fig. 5.7 we can observe a distribution whose mean and mode are biased to negative values, indicating improvements when substituting the SPLDA model by the PLDAUP. Besides, the skewness of the distribution is also negative, showing more relevance for negatives values of $\Delta\mathrm{DER}$ where PLDAUP outperforms SPLDA.

A similar study can be performed with cluster and speaker impurities. Fig. 5.8 repeats the preceding procedure, although now we evaluate both impurities instead of DER. Similar histograms analyzing impurities for the pool of partitions are shown, differentiating between cluster (Fig. 5.8a) and speaker (Fig. 5.8b) impurities.

The results in Fig. 5.8 show distributions with mean and mode close to zero for both cases. Differences arise when considering higher order moments, such as skewness when both impurities have opposite behaviour (speaker impurity is negative while cluster impurity is positive), and kurtosis, being higher in the cluster impurity distribution. The behaviour observed

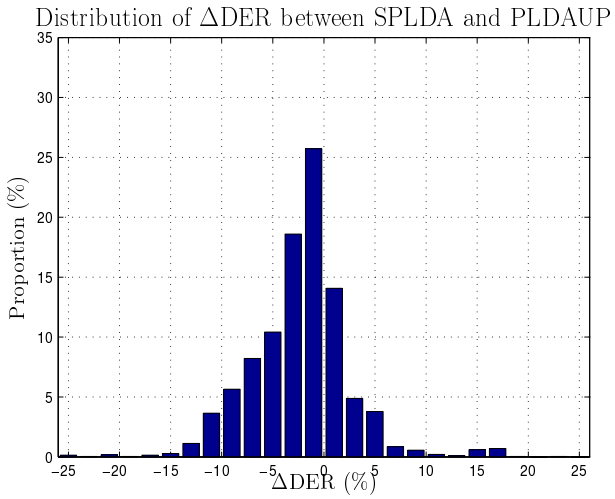
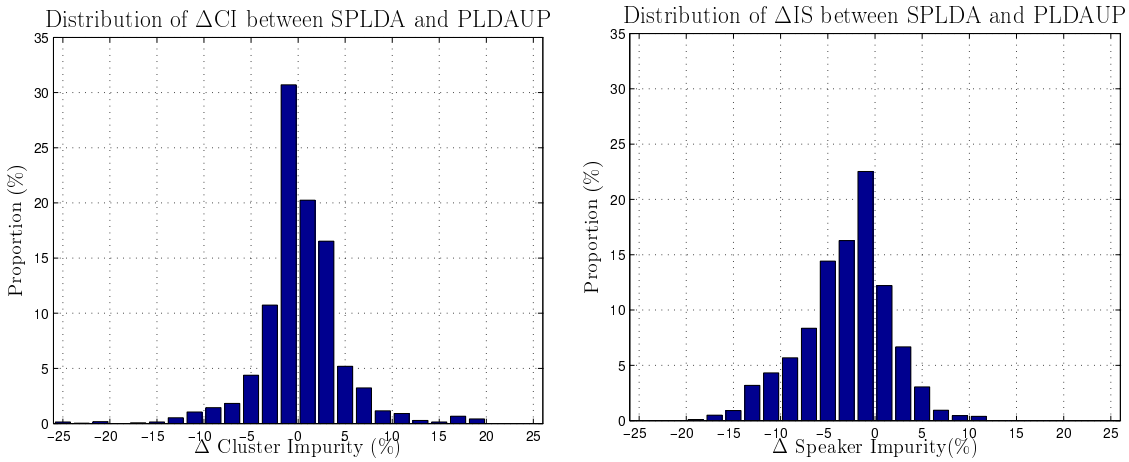


Figure 5.7: Histogram of DER variations between SPLDA and PLDAUP in MGB 2015 data.



a) Cluster Impurity b) Speaker Impurity

Figure 5.8: Histogram of a) cluster and b) speaker impurities variations between SPLDA and PLDAUP initializations in MGB 2015 data.

Experiment	DER(%)
SPK Setup	
40	40.61
50	39.83
75	**39.72**
Best FBPLDA	41.37
ELBO Setup	
ELBO	39.83
Best FBPLDA	**39.12**

Table 5.3: DER (%) results with the FBPLDAUP model in MGB 2015.

in Fig. 5.8 does not match with those previously obtained in Section 5.2.2. While in those experiments benefits were obtained in the cluster impurity, remaining speaker impurity almost unaltered, in broadcast data cluster impurity shows an average 1.24% absolute extra degradation, with 70% of the partitions degrading this metric, and speaker impurities show a 2.05% absolute improvement, common for 63% of the partitions.

A conclusion extracted from these results indicates that PLDAUP does no longer provide such improvements obtained in telephone channel experiments. Some causes for this degradation lie on the effect of short utterance training. In the telephone channel experiments we observed how short utterances trials were better evaluated as long as the SPLDA model considered them during training. By contrast, PLDAUP did not show any improvement but small degradations when trained with short utterances. In our current scenario we must deal with very short utterances, much shorter than those used in telephone channel experiments. Hence some reduction of the expected improvements seems reasonable.

The final step is the inclusion of the new model FBPLDAUP on top of the initialization, carried out by AHC with a PLDAUP model. In Table 5.3 we include those experiments with the new model architecture for MGB 2015 evaluation subset. Two different criteria of hypothesis selection have been evaluated: prior speaker number estimation and ELBO choice. The experiments include those results obtained with the new approach as well as a line indicating those results previously obtained with the non-UP models.

According to the results shown in Table 5.3, the FBPLDAUP shows potential benefits despite final results are overcome by traditional FBPLDA. When comparing setups with a fixed number of speakers our results clearly improve those obtained by FBPLDA. However, our new model does not get any benefit from the ELBO stop criterion while FBPLDA does. A possible

Experiment	DER(%)
SPK Setup	
30	18.16
40	**18.01**
50	18.66
Best FBPLDA	18.36
ELBO Setup	
ELBO	18.66
Best FBPLDA	**17.77**

Table 5.4: DER (%) results with the FBPLDAUP model in Albayzín 2018.

model	Real time factor
FBPLDA	0.01
FBPLDAUP	1.03

Table 5.5: Time costs for FBPLDA and FBPLDAUP with fixed number of speakers.

explanation for this situation is the lack of robustness in the ELBO estimation, which makes use of estimations for latent variables ($\mathbf{x}_{ij}$) considering only one utterance.

A similar study can be carried out with Albayzín 2018 dataset. The obtained results are illustrated in Table 5.4.

Table 5.4 shows similar trends as those previously seen with MGB 2015. When a fixed number of speakers is a forced, FBPLDAUP is able to improve FBPLDA results (now just slightly). However, FBPLDAUP again fails to improve when the ELBO partition choice criteria is applied.

The obtained results in both datasets have shown the potential benefits of this approach. Nevertheless, the inclusion of this concept by the way it was done has demonstrated some limitations in the partition comparison. Moreover, the computational cost must also be taken into account. The uncertainty propagation requires the use of particular matrices for each utterance $\mathbf{U}_j$, which makes impossible any precalculation. Additionally, the way $\mathbf{U}_j$ are defined force their inclusion in the slowest steps of the diarization: matrix inversions. This excess of computations affects both the initialization, done by means of PLDAUP, as well as during the resegmentation. Hence a time analysis is included in Table 5.5 to fairly compare both FBPLDA and FBPLDAUP options:

According to Table 5.5, FBPLDA is approximately 100 times faster than its version with

uncertainty propagation. Thus, if other approaches are interested in this concept, alternative strategies for the inclusion of the uncertainty should be considered in order to be more efficient.

5.5 Conclusions

Along this chapter we have evaluated an updated version of the FBPLDA model described in Chapter 4, now including the uncertainty propagation concept. This approach tries to exploit the knowledge compressed in the i-vector precision matrix, up to this point not considered in diarization.

During all our experiments in telephone channel incorporating the uncertainty propagation we observed benefits of the new approach. Despite being limited in 1vs1 evaluations, those experiments in speaker clustering provided promising results. Nevertheless, we also registered that the uncertainty propagation concept gets better when model training considers long utterances rather than short ones.

Moving towards broadcast data, clustering experiments with uncertainty propagation also showed improvements in performance, although not as relevant as in telephone channel. According to the experimental conditions, we consider that broadcast diarization datasets contain much more shorter segments, among other factors, influence this reduction of performance. When considering uncertainty propagation incorporated to the FBPLDA model, the FBPLDAUP model, we observed that it is able to improve the diarization performance as long as a similar stop criterion is applied. Nevertheless, an ELBO-based stop criterion did not work at all, compared to FBPLDA. In fact, the benefits for ELBO stop criterion is what makes FBPLDA outperform the FBPLDAUP version.

Finally, an efficiency analysis indicates that uncertainty propagation makes clustering a very slow task. This is a consequence of how the i-vector precision information is included into the PLDA-based model. Therefore, any future system reexploring this line of research should better develop a more effective addition.

Chapter 6

Tree-Based Clustering Approaches

This chapter is dedicated to alternative clustering approaches different to the Fully Bayesian PLDA clustering approach, considered in Chapters 4 and 5. This technique, according to our experiments, has clearly outperformed the results obtained with our reference AHC system. The possibility of simultaneously reassign all labels according to their relationships is a very powerful tool. Unfortunately, this clustering approach is far from perfection.

Its need for a proper initialization is its main flaw. Under the assumption that each cluster mainly contains audio from a single speaker and most of the embeddings from a speaker belong to the same cluster, FBPLDA resegmentation (with and without Uncertainty Propagation) would not degrade clear decisions while fixing errors with spurious embeddings. Unfortunately, if the initial partition happens to be mistaken, e.g. a speaker is split in different modes or a mode contains multiple speakers, the VB reclustering is usually unable to fix it. Thus, we ideally need a clean initial partition for a proper reclustering. However, this clean partition is the diarization solution we are looking for.

Alternatively, we can analyze clustering from a different perspective, treating it as a decoding problem, represented by a tree structure. This tree representation can be exploited by means of multiple strategies, including solutions using the product rule of probability concept. In the following lines we first present our complementary vision about clustering, explaining the reasons for the tree modeling. Later on, we will describe a solution based on this tree structure. The experimentation for this strategy will be explained afterwards. Finally, we include our conclusions about this new point of view.

6.1 Tree-based point of view for clustering

In Chapter 2 we defined clustering in diarization as the task dedicated to infer the set of labels $\Theta = \{\theta_1, ..., \theta_j, ...\theta_N\}$ for the set of N embeddings $\mathbf{\Phi} = \{\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_j, ..., \boldsymbol{\phi}_N\}$ in such a way that all representations in $\mathbf{\Phi}$ from the same speaker are labeled together. Moreover, we can define $\Omega_\Theta = \{\Theta^1, ..., \Theta^m, ..., \Theta^{B_N}\}$ as the set of possible partitions in which the elements in $\mathbf{\Phi}$ can be distributed, being B_N the Bell number for N elements. According to our understanding, Ω_Θ can be represented by means of a tree structure that we denote as clustering tree. This tree can be recursively defined from the leaves, where the different partitions lie, up to the root. The intermediate nodes represent partial partitions of $\mathbf{\Phi}$, containing more and more embeddings as long as the node is placed deeper in the tree. The definition is:

T is a clustering tree for a disjoint subset $\Omega_{\Theta_T} \in \Omega_\Theta$ if it either represents a single partition $T = \{\Theta^m\} : m = 1..B_N$ or denotes a fusion of clustering subtrees $T = \{T_1, ..., T_t, ..., T_{nT}\}$. Besides, all T_t subtrees of T must have the same depth D, each one representing all partitions from Ω_Θ whose labels only differ in the last $D-1$ values. In consequence, the clustering tree T describing Ω_Θ is unique, presenting a depth N for the B_N leaves. In this tree each node is responsible for the representation of a partial partition $\Theta'_D \in \Theta$, where Θ'_D corresponds to the partial labeling for the node at depth D. This partial assignment $\Theta'_D = \{\theta_1, ..., \theta_D\}$ consists of the first D labels of Θ. Hence, while the node at the root of the tree T should explain the trivial partition $\Theta'_1 = \{\theta_1\}$, each leaf at depth N should explain the whole partition $\Theta = \{\theta_1, ..., \theta_j, ..., \theta_N\}$. Therefore, the tree T', representing the clustering of a subset $\mathbf{\Phi}' \in \mathbf{\Phi}$ of D embeddings $\mathbf{\Phi}' = \{\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_D\}$, shares the same tree structure of the tree T up to depth D.

This tree T for Ω_Θ perfectly matches a sequential clustering where elements in $\mathbf{\Phi}$ are sequentially assigned to the already existing clusters or made responsible for a new group. An example for this clustering tree only considering four elements is shown in Fig. 6.1, where the number in each node at depth j denotes the value for the label θ_j for all partitions of the subtree.

The definition of clustering from a tree perspective provides a different point of view about how clustering solutions infer the best partition. Many clustering strategies, including those used in previous chapters of this thesis (AHC, VB, etc.), only move through the tree leaves, taking into account an initial partition which evolves along multiple steps. However, apart from this purely horizontal traversal we can also perform a vertical one, decoding a path through the tree structure that connects the root to the leaf with the target partition.

By considering clustering as a decoding task on a tree structure, many tools can be born in mind. One of the most popular strategies to deal with trees is the Viterbi algorithm

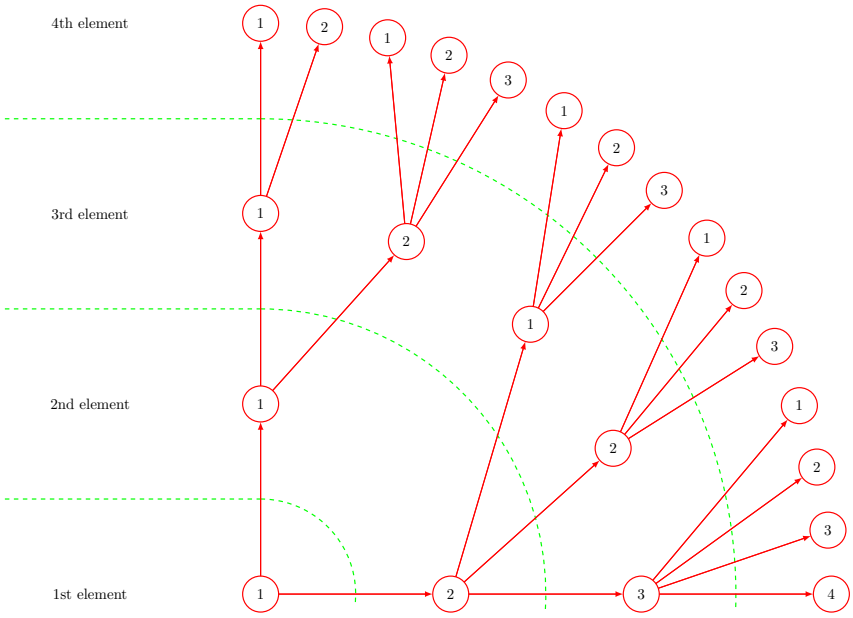


Figure 6.1: 4-level tree clustering example. The number in each node denotes the value for the label θ_n along all represented partitions of the subtree

[Viterbi, 1967], which, under certain conditions, can find the optimal path through a tree structure. However, Viterbi requires as necessary condition that any subtree T_t should present its identical structure in multiple divergent paths. By means of this condition the Viterbi algorithm exploits memoryless properties about precedent decisions, which allows the accommodation of the tree structure in an efficient lattice representation. However, the definition of clustering tree does not satisfy Viterbi necessary condition and thus suboptimal solutions must be considered instead.

The M-Algorithm [Jelinek and Anderson, 1971] is a suboptimal proposal to carry out an efficient search of the best path along a tree while taking into account the tradeoff between exploration of paths and computational costs. This tradeoff is maintained by a restricted exploration of the tree, only analyzing a set of M paths likely to be the optimal solution.

The optimization criteria must be defined at the same time we choose the quality metric it must work with. This metric is a measure to determine how well each partition fits the data in $\mathbf{\Phi}$. Depending on the clustering solution, this metric should only need to evaluate the nodes of the tree at depth N where partitions lie. Nevertheless, vertical traversals along the clustering tree T require evaluating nodes at intermediate depths. For this purpose, statistical approaches provide excellent tools which perfectly match with tree structures. These tools, based on the product rule of probability, have been exploited to design the PLDA tree-based clustering [Viñals et al., 2019a].

6.2 PLDA tree-based clustering

The PLDA tree-based clustering is a statistical generative solution to the diarization clustering task. Hence, it follows the principle described in Section 2.6.2, identifying the target partition Θ for the set of embeddings $\mathbf{\Phi}$ as the one maximizing $P(\mathbf{\Phi}, \Theta)$. In order to do so it exploits the tree perspective of the clustering problem and the path decoding strategies.

The application of statistical strategies on top of a tree structure associates a probability to each node. Regarding the leaves this probability is $P(\mathbf{\Phi}, \Theta_m) : m = 1..B_N$, i.e. the quality metric for each partition. In order to obtain a similar probability for the remaining nodes we make use of the product rule of probability. This rule allows the decomposition of a generic $P(a_1, ..., a_N)$ as:

$$\begin{aligned} P(a_1, ..., a_N) =& P(a_1)\, P(a_2|a_1)\, P\left(a_3|a_1^2\right) ... P\left(a_N|a_1^{N-1}\right) \\ =& \prod_{j=2}^{N} P\left(a_j|a_1^{j-1}\right) P(a_1) \end{aligned} \tag{6.1}$$

where a_1^j represents the set of elements $\{a_1, ..., a_j\}$. This definition can also be expressed in a recursive way:

$$P\left(a_1^j\right) = P\left(a_j|a_1^{j-1}\right) P\left(a_1^{j-1}\right) \tag{6.2}$$

The application of the product rule of probability to our diarization problem is direct, substituting the jth element a_j from the previous equation by the jth pair of variables, consisting of the embedding $\boldsymbol{\phi}_j$ and its cluster identity label θ_j. Hence, the probability for any partition $P(\mathbf{\Phi}, \Theta)$ can be decomposed as:

$$P(\mathbf{\Phi}, \Theta) = \prod_{j=2}^{N} P\left(\boldsymbol{\phi}_j, \theta_j|\boldsymbol{\phi}_1^{j-1}, \theta_1^{j-1}\right) P(\boldsymbol{\phi}_1, \theta_1) \tag{6.3}$$

and its alternative recursive definition:

$$P\left(\boldsymbol{\phi}_1^j, \theta_1^j\right) = P\left(\boldsymbol{\phi}_j, \theta_j|\boldsymbol{\phi}_1^{j-1}, \theta_1^{j-1}\right) P\left(\boldsymbol{\phi}_1^{j-1}, \theta_1^{j-1}\right) \tag{6.4}$$

In consequence, each node at depth j in the clustering tree T has the probability $P\left(\boldsymbol{\phi}_1^j, \theta_1^j\right); \theta_1^j \in \Omega_{\theta_1^j}$ associated. According to this decomposition we are assuming $\mathbf{\Phi}$ as a sequence of ordered embeddings to be clustered. Besides, these embeddings only depend on previous speaker representations of the sequence. These assumptions are reasonable in real life, where the voice evolves along time, being specially noticeable in large segments of speech.

6.2.1 PLDA-based model

Along the previous lines we explored a new perspective about clustering, representing it as a tree structure to be decoded in order to obtain the best partition. Moreover, the statistical strategy associated a probability $P\left(\boldsymbol{\phi}_1^j, \theta_1^j\right)$ to all nodes along the tree. However, the distribution for this probability has not been specified yet. Considering speaker recognition state of the art, PLDA family models seem a powerful type of solution to apply.

Thus, we must keep on transforming $P\left(\boldsymbol{\phi}_1^j, \theta_1^j\right)$ to make PLDA definition applicable. As a first transformation, we keep on applying the product rule of probability, decomposing the probability at each node into a term depending on the embeddings, the conditional distribution, and a prior distribution of the labels. This decomposition is:

$$P\left(\boldsymbol{\phi}_j, \theta_j | \boldsymbol{\phi}_1^{j-1}, \theta_1^{j-1}\right) = P\left(\boldsymbol{\phi}_j | \theta_j, \boldsymbol{\phi}_1^{j-1}, \theta_1^{j-1}\right) P\left(\theta_j | \boldsymbol{\phi}_1^{j-1}, \theta_1^{j-1}\right) \tag{6.5}$$

This decomposition allows to split $P\left(\boldsymbol{\phi}_1^j, \theta_1^j\right)$ into two simpler problems. Now, we exclusively focus on the first term, the conditional distribution of the embedding j given its jth label as well as previous embeddings $\boldsymbol{\phi}_1^{j-1}$ and labels θ_1^{j-1}. Decisions about the other term, the prior distribution of the current label θ_j given previous embeddings and decisions will be made afterwards. Unfortunately, this conditional term is still intractable to use PLDA due to the presence of label variables. Moreover, PLDA only defines dependencies among embeddings from the same speaker. Therefore, our next transformations seek separating embeddings from the labels. We take inspiration from Chapter 4, imposing $P\left(\boldsymbol{\phi}_j | \theta_j, \boldsymbol{\phi}_1^{j-1}, \theta_1^{j-1}\right)$ to follow a multinomial distribution on the variable θ_j, a one-hot sample with I values ($\theta_j = \{\theta_{1j}, ..., \theta_{ij}, ..., \theta_{Ij}\}$), where I is the number of candidate speakers. Thus, the value θ_{ij} will take the value of one if the jth embedding was generated by the speaker i, being zero otherwise. Besides, we also require $\boldsymbol{\phi}_j$, when belonging to cluster i according to θ_j, to be exclusively explained by those embeddings already assigned to this cluster. This subset of embeddings previously assigned to cluster i at time j is denoted by $\boldsymbol{\Phi}_{ij}$. Under these to conditions we can express:

$$P\left(\boldsymbol{\phi}_j | \theta_j, \boldsymbol{\phi}_1^{j-1}, \theta_1^{j-1}\right) = \prod_{i=1}^{I} P\left(\boldsymbol{\phi}_j | \boldsymbol{\Phi}_{ij}\right)^{\theta_{ij}}; j = 1..N \tag{6.6}$$

The definition of the term $P\left(\boldsymbol{\phi}_j | \boldsymbol{\Phi}_{ij}\right)$ now makes the application of PLDA principles feasible. First, we must assume the existence of a latent variable representing the speaker information $\mathbf{y}_i$, which allow us to redefine $P\left(\boldsymbol{\phi}_j | \boldsymbol{\Phi}_{ij}\right)$ as:

$$P\left(\boldsymbol{\phi}_j | \boldsymbol{\Phi}_{ij}\right) = \int P\left(\boldsymbol{\phi}_j | \mathbf{y}_i\right) P\left(\mathbf{y}_i | \boldsymbol{\Phi}_{ij}\right) d\mathbf{y}_i \tag{6.7}$$

Given this definition we can now assume that the data we are dealing with is generated by a PLDA model. For this purpose, we make use of the SPLDA conditional distribution $P\left(\boldsymbol{\phi}_j|\mathbf{y}_i, \mathcal{M}_{\text{SPLDA}}\right)$:

$$P\left(\boldsymbol{\phi}_j|\mathbf{y}_i, \mathcal{M}_{\text{SPLDA}}\right) \sim \mathcal{N}\left(\boldsymbol{\phi}_j|\boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i, \mathbf{W}^{-1}\right) \tag{6.8}$$

where $\boldsymbol{\mu}$ is the speaker independent term, $\mathbf{V}$ a low rank matrix describing the speaker subspace and $\mathbf{W}$ a full rank matrix explaining the intra-speaker variability space.

Furthermore, the second term $P\left(\mathbf{y}_i|\boldsymbol{\Phi}_{ij}\right)$, the posterior distribution of the latent variable given all those embeddings previously assigned to cluster i ($\boldsymbol{\Phi}_{ij}$) is also modeled according to SPLDA. Thus, its definition is:

$$P\left(\mathbf{y}_i|\boldsymbol{\Phi}_{ij}, \mathcal{M}_{\text{SPLDA}}\right) \sim \mathcal{N}\left(\mathbf{y}_i|\boldsymbol{\mu}_{\mathbf{y}_i}(j), \mathbf{L}_{\mathbf{y}_i}^{-1}(j)\right) \tag{6.9}$$

$$\mathbf{L}_{\mathbf{y}_i}(j) = \mathbf{I} + \mathbf{V}^T \sum_{k=1}^{j-1} \theta_{ki} \mathbf{W}\mathbf{V} \tag{6.10}$$

$$\boldsymbol{\mu}_{\mathbf{y}_i}(j) = \mathbf{L}^{-1}\mathbf{V}^T\mathbf{W} \sum_{k=1}^{j-1} \theta_{ki}(\phi_k - \boldsymbol{\mu}) \tag{6.11}$$

where $\boldsymbol{\mu}_{\mathbf{y}_i}(j)$ and $\mathbf{L}_{\mathbf{y}_i}(j)$ represent the estimates for the mean and variance parameters respectively of the latent variable $\mathbf{y}_i$ when only $j-1$ elements were observed. As long as j increases these estimations should get closer to the real value.

Apart from well-known definitions for both distributions, the choice of the SPLDA model also provides an extra advantage. Its Gaussian nature for both $P\left(\boldsymbol{\phi}_j|\mathbf{y}_i, \mathcal{M}_{\text{SPLDA}}\right)$ and $P\left(\mathbf{y}_i|\boldsymbol{\Phi}_{ij}, \mathcal{M}_{\text{SPLDA}}\right)$ allows a closed form solution to the integral defining $P\left(\boldsymbol{\phi}_j|\boldsymbol{\Phi}_{ij}, \mathcal{M}_{\text{SPLDA}}\right)$. The resulting formulation for this term is:

$$P\left(\boldsymbol{\phi}_j|\boldsymbol{\Phi}_{ij}, \mathcal{M}_{\text{SPLDA}}\right) \sim \mathcal{N}\left(\boldsymbol{\phi}_j|\boldsymbol{\mu}_i(j), \Sigma_i(j)\right) \tag{6.12}$$

$$\boldsymbol{\mu}_i(j) = \boldsymbol{\mu} + \mathbf{V}\boldsymbol{\mu}_{\mathbf{y}_i}(j) \tag{6.13}$$

$$\Sigma_i(j) = \mathbf{W}^{-1} + \mathbf{V}\mathbf{L}_{\mathbf{y}_i}^{-1}(j)\mathbf{V}^T \tag{6.14}$$

After completely defining the conditional distribution of the embeddings, we now can pay attention to the label prior distribution $P\left(\theta_j|\boldsymbol{\phi}_1^{j-1}, \theta_1^{j-1}\right)$. First, we assume a simplified prior distribution by eliminating the dependence with respect to the past embeddings $\boldsymbol{\phi}_1^{j-1}$. Therefore, our prior distribution will follow the form $P\left(\theta_j|\theta_1^{j-1}\right)$. For the resulting distribution we have opted for the Distance Dependent Chinese Restaurant (DDCR) process [Blei and Frazier, 2011], already used in diarization in [Zhang et al., 2019]. This model explains the occupation of an

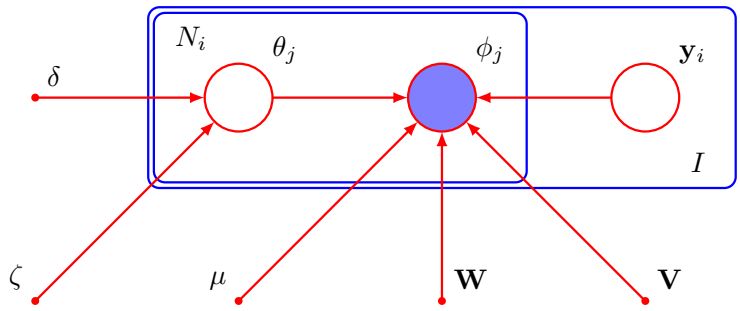


Figure 6.2: PLDA tree-based clustering Bayesian Network

infinite series of clusters by a sequence of elements. Then, the assignment of the element j to any cluster exclusively depends on the occupation of clusters up to this point, i.e. according to all the previous decisions, as in our decomposition. The probability of assignment of the element j to any of the already created $k = 1..K$ clusters is proportional to its occupation at time j, namely n_k. Besides, DDCR offers the possibility to create a new cluster $K+1$ proportional to γ. The mathematical formulation for DDCR is:

$$P\left(\theta_j = k|\theta_1^{(j-1)}\right) \propto \begin{cases} n_k & \text{if } k \leq K \\ \zeta & \text{if } k = K+1 \end{cases} \tag{6.15}$$

DDCR deeply matches sequential ordering and assignment problem. Unfortunately, DDCR considers reasonable a continuous transition among speakers. Applied to scenarios of speaker clustering, where speaker recording can be interleaved, seems reasonable. However, in diarization we must consider the segmentation stage, which can divide any long segment into pieces of shorter length. Thus, we can add to this distribution more chances to remain in the speaker cluster. In our proposal we do so by specifically defining the situation of remaining in the current speaker cluster, with a probability proportional to δ. This addition generates the following modification of the DDCR distribution:

$$P\left(\theta_j = k|\theta_1^{(j-1)}\right) \propto \begin{cases} \delta & \text{if } k = \theta_{(j-1)} \\ n_k & \text{if } k \neq \theta_{(j-1)} \text{ and } k \leq K \\ \zeta & \text{if } k \neq \theta_{(j-1)} \text{ and } k = K+1 \end{cases} \tag{6.16}$$

The model $P(\mathbf{\Phi}, \Theta)$, taking into account the whole set of assumptions previously described, can be represented by the Bayesian network illustrated in Fig. 6.2

6.2.2 M-algorithm optimization

Once the PLDA-based model is defined, now it is time to find the way to obtain those labels Θ_{diar} that best explain the set of embeddings $\mathbf{\Phi}$. Taking into account that the Viterbi algorithm

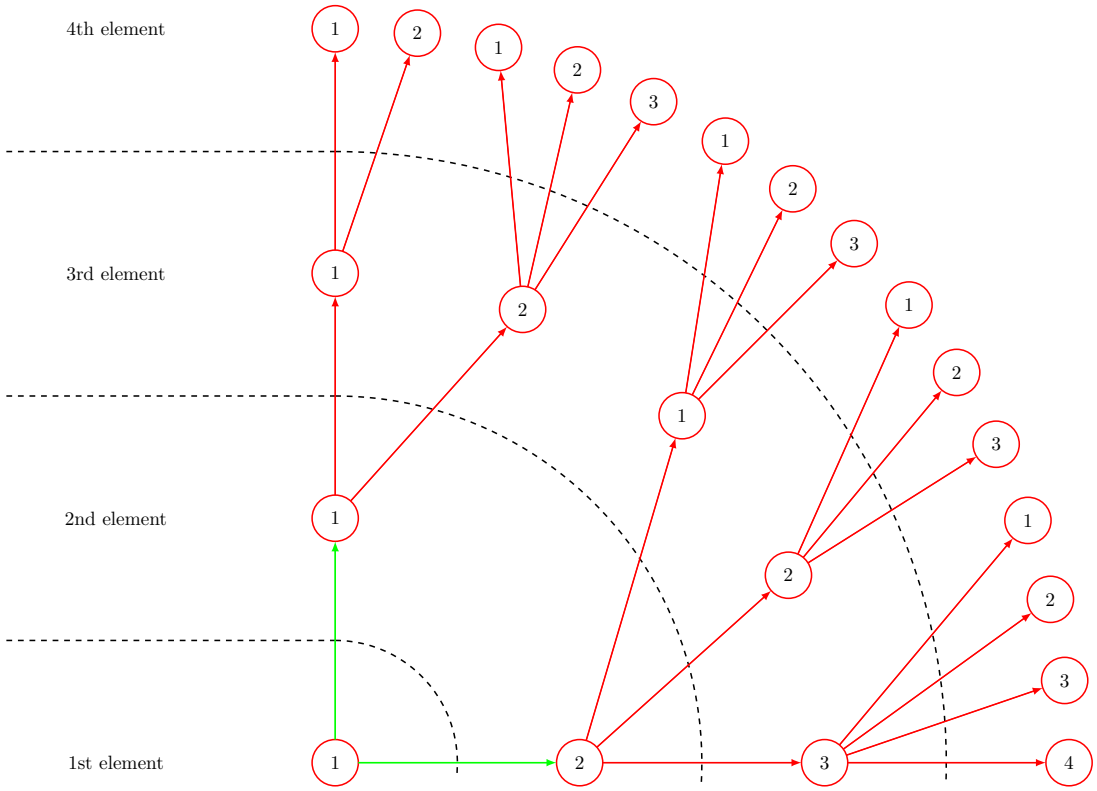


Figure 6.3: M-algorithm example for a clustering tree of depth 4. 2 paths alive reach the depth 2 through the tree (green).

cannot be applied, suboptimal approaches considering trustworthy paths, as the M algorithm [Jelinek and Anderson, 1971] are still applicable.

The M algorithm is an iterative solution strategy. Given a scenario with a decision tree of depth N, the M algorithm tracks a subset of M surviving paths, i.e. those paths more likely to be the solution (in our case those with higher log-likelihood). Besides, all paths must have reached depth j within the tree structure. Thus, the goal is the identification of those best transitions taking the M tracked paths from depth j to depth $j + 1$. In Fig. 6.3 we illustrate an example, where a clustering tree of depth 4 is analyzed by the M algorithm with $M = 2$. Surviving path (green lines) have reached depth 2 within the tree.

The M algorithm iterative procedure is divided into two steps, estimation and maximization. The estimation step studies how the M surviving paths in level j evolve deeper through the tree, predicting its performance in a future scenario and making decisions in consequence. For this purpose we carry out a brute-force approach, analyzing any possible transition from the M surviving paths at depth j up to a certain extra depth d. The parameter d is a design choice and responsible for a tradeoff between accuracy and computational costs. The higher d, the wider

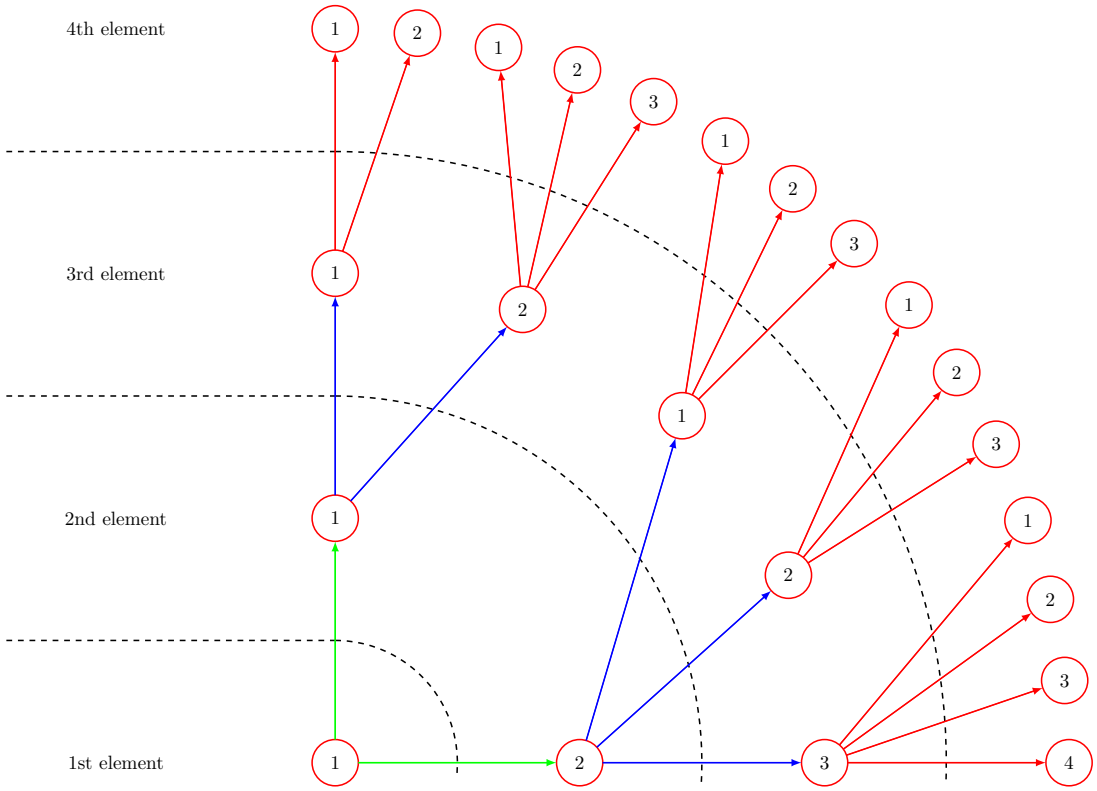


Figure 6.4: Estimation step in a M-algorithm example for a clustering tree of depth 4. 2 paths alive (green) reaching depth 2 are propagated to all possible nodes at depth 3 (blue)

is the exploration of the tree for unseen data and hence higher accuracy might be expected, but increasing in an exponential manner the computational costs. Hence, many systems restrict d to be equal to 1. In Fig. 6.4 we represent the estimation step applied to our previous example scenario in Fig. 6.3. Each surviving path (green line) is propagated d ($d = 1$) levels ahead (blue lines), evaluating for each configuration the performance at this depth.

The results of the estimation step provide an overview about how the tree behaves in future steps, without compromising any decision. This choice is made during the maximization step. In this step all candidate propagations are ranked, only keeping those M with better score. These new M paths now reaching depth $j + 1$ are our most promising candidates so far, and those considered for the next iteration of the algorithm. This step is represented in Fig 6.5.

6.3 Experiments

For the evaluation of the new clustering approach, we will make use of Albayzín 2018, as described in Section 3.2.2. For this purpose, we consider an i-vector PLDA diarization system

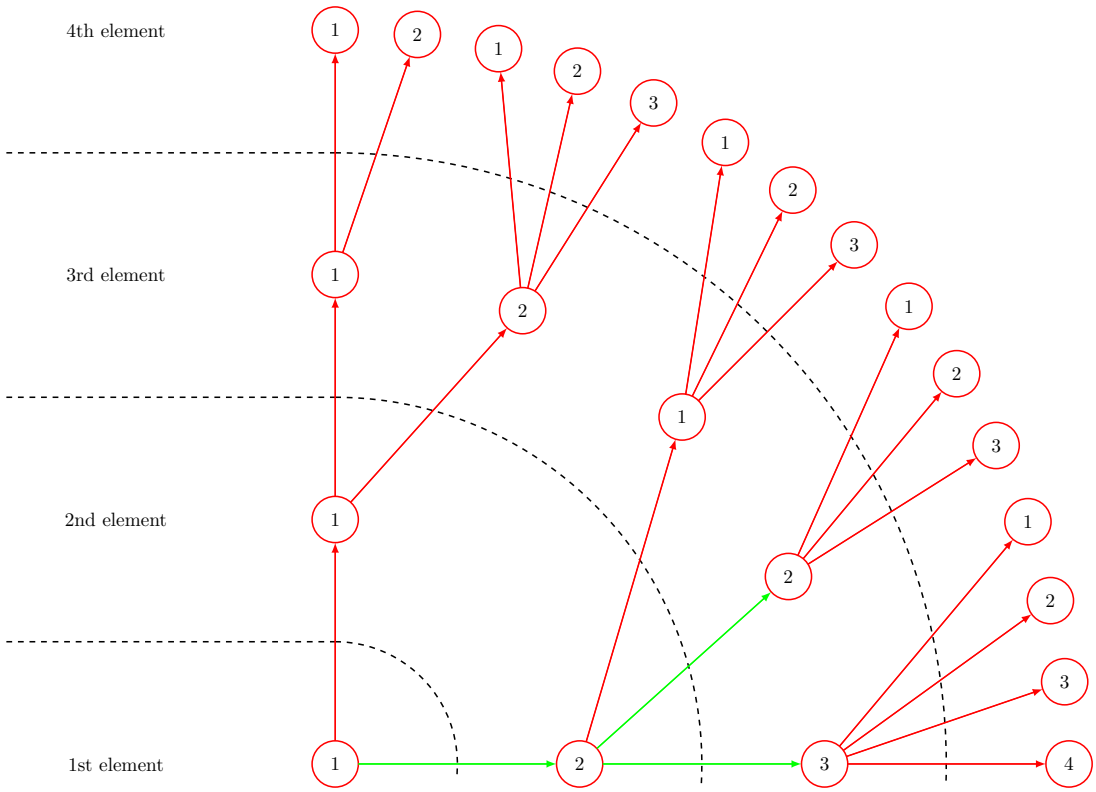


Figure 6.5: Maximization step in a M-algorithm example for a clustering tree of depth 4. The two surviving paths are shown in green.

Experiment	DER(%)	
	Dev. Subset	Eval. Subset
AHC	18.88	26.36
AHC + FBPLDA	13.90	17,79
PLDA TREE-BASED CLUSTERING	**13.12**	**17.60**

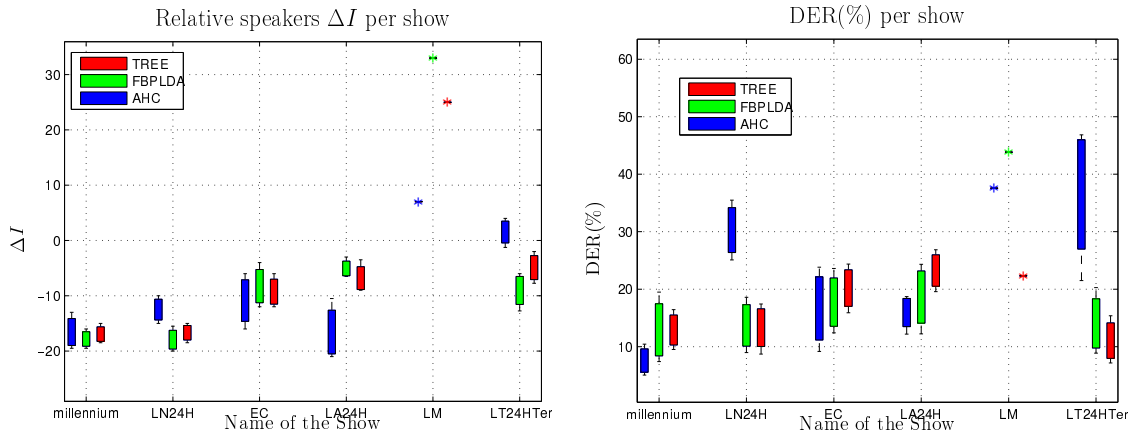
Table 6.1: DER (%) results for the PLDA tree-based clustering in Albayzín 2018. Results compared with those obtained by means of AHC with and without FBPLDA resegmentation.

whose setup is: A 256 Gaussian GMM-UBM followed by a 100-dimension Total Variability matrix are responsible for the i-vector extraction. The obtained embeddings undergo centering, whitening and length normalization prior to clustering, without dimensionality reduction. Finally, the new clustering approach, the PLDA tree-based clustering, uses a 100-dimension SPLDA. This setup fits in terms of dimensions with the diarization system using the FBPLDA reclustering in Chapter 4 for experiments with Albayzín 2018.

In our first experiment we compare the performance of FBPLDA reclustering, obtained in Chapter 4, and our new clustering approach. As a first approximation we assume the set of embeddings $\mathbf{\Phi}$ to be arranged in temporal order. We restrict hyperparameter d to be equal to 1 for computational reasons. In this experiment we consider evaluation conditions, i.e. we only present the performance of the best hyperparameter configuration (δ, ζ and M) according to Albayzín 2018 development subset. The obtained results are shown in Table 6.1.

According to the obtained results, the new clustering approach, working with i-vectors, provides very little improvement with respect to the FBPLDA counterpart. However, these results show benefits in the evaluation of both development and test subsets despite containing independent shows. Therefore, we can talk about limited yet consistent improvements due to our new clustering approach.

Apart from the overall score for both development and test subsets, a more detailed analysis of results can also be done. For this purpose, we study the performance per show of interest with the three clustering approaches considered along this thesis: AHC, FBPLDA and PLDA tree-based clustering. For this purpose, we analyze two different metrics: On the one hand we propose the analysis of $\Delta I = I_{\mathrm{ORACLE}} - I_{\mathrm{HYP}}$, the difference in the number of speakers between our hypothesis labels and the reference. On the other hand, we analyze the DER performance. Both analyses are shown in Fig. 6.6, including all shows in Albayzín 2018. The involved shows from the development subset are *millenium* and *La Noche en 24 Horas* (LN24H). Regarding the test subset, the shows *España en Comunidad* (EC), *Latinoamérica en 24 Horas* (LA24H), *La*



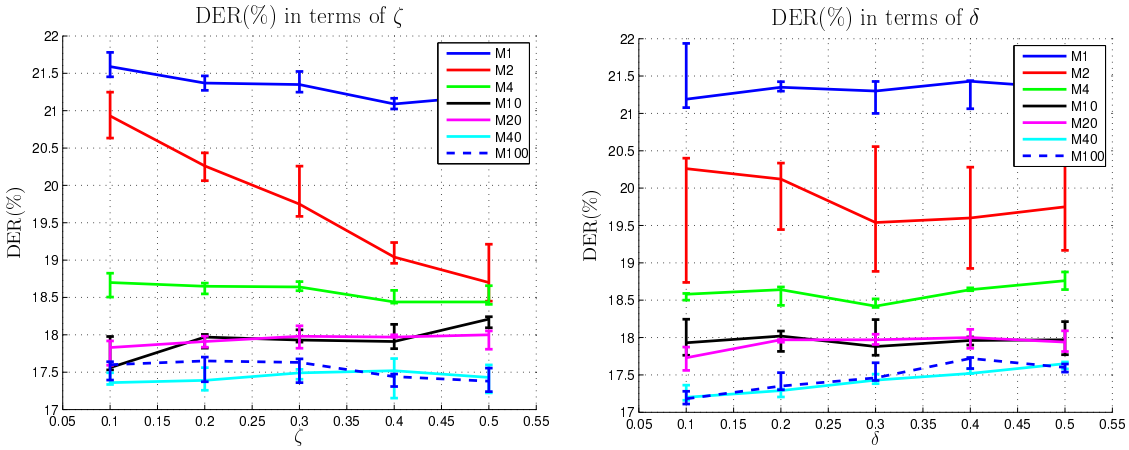
a) ΔI b) DER(%)

Figure 6.6: Analysis per show of a) ΔI and b) DER(%) for AHC, FBPLDA and PLDA tree-based clustering. Analysis carried out on shows from Albayzín 2018, including development and test subsets.

Mañana (LM) and *La Tarde en 24 Horas Tertulia* (LT24HTer) are also included. Results reflect the interquartile range for each show.

Those results illustrated in Fig. 6.6 show a similar behaviour of the three types of clustering per show. Thus, those more harmful shows are common for all systems. However, our new clustering approach shows a minor interquartile range per show compared to AHC and specially FBPLDA. This reduction affects both the estimation about the number of speakers and DER. Hence the performance of the system seems more consistent per individual show or domain, although small degradations might occur. This behaviour can also be extrapolated to the whole dataset, specially considering the show *La Mañana* (LM). While AHC and FBPLDA performances for this show are at least 100% worse than any other show in terms of DER, the PLDA tree-based clustering achieves to behave as bad as the second worst show. This improvement is also observed in the estimation of the speaker number, with a relative 25% degradation reduction.

Apart from a specific setup, we can also do an analysis studying the influence for each of the model hyperparameters δ, ζ and M. For this analysis we will consider the obtained scores for any possible setup. Fig. 6.7 is our chosen graphical representation to reveal the impact for the different hyperparameters. It is composed of two parts, Fig. 6.7a where we represent the relationship between ζ and DER for different values of M, and Fig. 6.7b , where we represent the relationship between δ, and DER for the different values of M. In order to include all hyperparameters in each subimage, Fig. 6.7a includes some variability per measure, illustrating the interquartile range results in terms of the missing hyperparameter, δ. Similarly, measures in



a) ζ probability b) δ probability

Figure 6.7: DER (%) results for the PLDA tree-based clustering with M-algorithm in Albayzín 2018 in terms of δ, ζ and M

Fig. 6.7b include some variability margins indicating the first and third quartile results in terms of ζ.

The information included in Fig. 6.7 reveals many important characteristics about the model. First, the results evidence the importance of M. 20% relative improvements may be obtained as long as more and more simultaneous paths are evaluated. However, this improvement is not uniform, being any increase of M more significant for lower values. For higher values of M, improvements are very scarce and implying large increments in the computational costs. Other detail to bear in mind is that, except for M hyperparameter, the influence of the remaining adjustable values (δ and ζ) is in general reduced (with the exception of ζ for $M = 2$). variations may be around 5% relative improvement/degradations, i.e. 1% absolute DER variations.

Up to this point we have only mentioned three existing hyperparameters, M δ and ζ. Nevertheless, all the results were obtained by setting the embeddings $\mathbf{\Phi}$ into a sequential order. If the analysis of the clustering tree was complete, i.e. analyzing each one of the leaves, the impact of this ordering would be null. Nevertheless, by partially exploring the clustering tree according to limited data makes this arrangement an extra factor to take into account. Therefore, while in our previous examples we exclusively applied temporal order, i.e. we can also apply different arrangements

One of the key factors when using this tree-based approach is the sequence order, specially taking into account that we are exploiting the relationships between an embeddings and its predecessors in the sequence. Thus, we must explore how the ordering affects the results. While in our previous experiments we simply made use of the temporal order, this arrangement is not

Experiment	DER
Time (Forward)	**17.60**
Time (Backward)	17.91
Random	21.32 ± 1.97
Segment length (Increasing)	**19.04**
Segment length (Decreasing)	21.02
Similarity (fine)	18.57
Similarity (coarse)	**17.43**

Table 6.2: DER (%) results of the PLDA tree-based clustering in terms of the embedding arrangement.

unique so alternatives can be proposed.

In order to analyze the impact of the arrangement in this technique, brute-force approaches are not suitable. For an episode of N embeddings, they can be ordered in $N!$ different arrangements. For this reason, we have evaluated a small subset of criteria:

- **Time**. Both forward and backward time.
- **Random**. A random ordering has been tested. 100 different arrangements have been evaluated, averaging the obtained results.
- **Segment length**. Embeddings reliability is conditioned by its length. Increasing and decreasing orders have been tested.
- **Similarity**. Embeddings are clustered according to PLDA pairwise log-likelihood ratio and a threshold. The ordering places embeddings from the same cluster in a consecutive order while maintaining its time order. Two criteria of clustering have been followed: On the one hand a fine criterion, where any pair of elements within the cluster must have a similarity overcoming the threshold. On the other hand, one element belongs to a cluster as long as it has a pairwise similarity with another element of the cluster over the threshold (coarse clustering).

The comparison of results has been carried out with Albayzín 2018 evaluation set. The hyperparameter tuning has not been modified to simplify the experimentation. The obtained results are shown in Table 6.2.

According to Table 6.2, we observe that the time ordering is a very powerful arrangement. For those examples where time order is partially maintained (Time and Similarity), performance

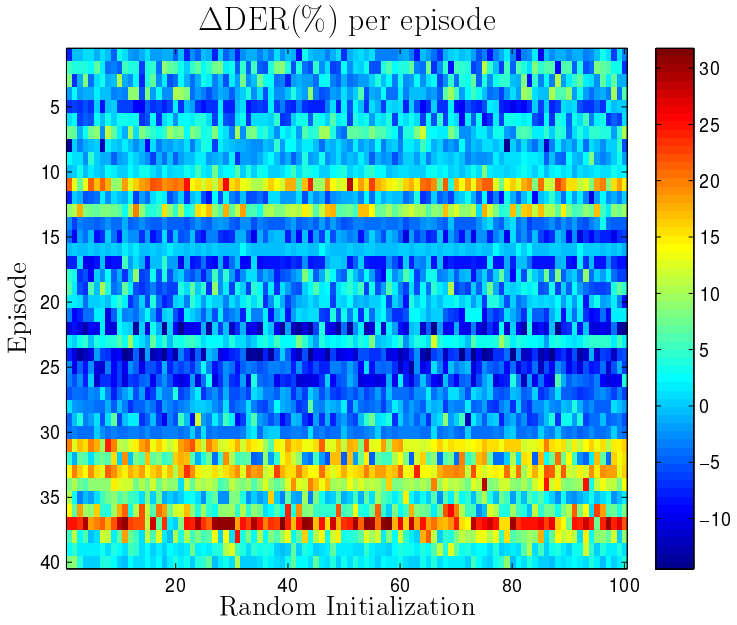


Figure 6.8: DER relative results between Random order and Time order

tends to be lower than in the other cases. This degradation is specially noticeable in the Random ordering, which obtains by far the worst results. It is significative that the segment length is not as important as time order, regardless the direction. Finally, the similarity ordering criteria seems to be more powerful when a coarse clustering is applied, slightly outperforming our time order. However, further analysis must determine if this ordering is better than the fine counterpart because it exploits better the similarity information or because it is more likely to maintain the time order.

Finally, our final study about ordering analyzes two of the most extreme arrangements episode by episode. While the previous analysis illustrates the general performance, a more detail look could provide more information. For this reason, we compare episode by episode the pool of Random order diarizations and compare them with those obtained by the increasing time order. In Fig. 6.8 we have represented the variations in DER ($\Delta\text{DER} = \text{DER}_{RANDOM} - \text{DER}_{TIME}$) for all the 100 different initializations, analyzing all the audios in Albayzín 2018 evaluation subset.

Results in Fig. 6.8 provide an average degradation of 1.67% absolute DER degradation per show when random arrangement is considered. However, most of this degradation is due to a small subset of episodes where time ordering obtains an outstanding performance compared to any Random initialization (over 10% ΔDER). By excluding these data, time ordering is then 0.07% ΔDER worse than random ordering.

6.4 Conclusions

Along this chapter we have proposed an alternative interpretation for the clustering task, representing it as a tree structure in which any node denotes the assignment of an embedding to a cluster. Thanks to this new point of view, a new bunch of alternatives can be born in mind, as the already presented solution.

Regarding this solution, its performance with i-vectors is slightly better than the FBPLDA system. Nevertheless, its performance is more consistent along the multiple shows, including both development and test subsets.

With respect to the hyperparameter choice, we must remark the impact of M, the amount of surviving paths in the M algorithm. This term is also responsible to control the tradeoff between tree exploration and computational cost. By simply considering few simultaneous hypotheses diarization errors significantly improve up to a relative 20%. Moreover, this improvement is not linear, being specially concentrated in the lowest values. Thus, while lower M values may provide a significant yet limited improvement with low latency and memory requirements, larger values of M may provide more refined results although requiring more time and computational resources.

Finally, our new proposed solution performs a partial exploration of the clustering tree so the element arrangement is a factor to take into account. According to our experiments, we have observed that time ordering is a reasonable option. This result a consequence of the evolution of the voice from a speaker along time, which in general can be considered smooth and thus beneficial for our system. However, broadcast content does not always present speakers in a single condition. Thus, time order may be outperformed by alternative arrangements simultaneously exploiting other types of variability in the speech.

Part III

The Speaker Representation Problem

Chapter 7

Study of embeddings for short utterances

During Chapters 4, 5 and 6 we have analyzed the clustering stage, one of the key blocks in diarization. However, the modelling capabilities of the embedding extraction is also a capital step in diarization. The higher their discriminative properties, the more effective is the posterior clustering. Thanks to the relationships between diarization and speaker recognition, state-of-the-art embeddings have historically been considered. Nevertheless, diarization segments usually present specific features that require some special treatment, as the short utterance problem. Thus, this chapter follows the ideas in [Viñals et al., 2019b], dedicated to a proper understanding about the typical embedding extraction procedures and their implications with the short utterance problem.

7.1 Introduction

Speaker recognition is the area of speech technologies that allows the automatic recognition of the speaker identity given some portions of his/her speech. Its goal is the proper characterization of the speaker, isolating singular characteristics of his/her voice and making possible accurate comparisons among different speakers.

Under certain circumstances (telephone channel 5-minute utterances) the current state of the art has achieved outstanding results, with error rates below 1% EER [Sadjadi et al., 2016]. In general, as long as we have enough speech to characterize a speaker, current representations are robust enough. By contrast, when utterances get shorter the performance of speaker recognition techniques is severely degraded, as shown in Section 5.2.1. This issue is gaining relevance because the short utterance scenario is becoming more and more common. Conversational speech is composed of interleaved relatively short contributions (1s-30s approximately depending on the domain) from the different speakers. The identification of these short contri-

butions, i.e. the diarization task, usually works with even shorter segments (1s-3s) to accurately deal with speaker boundaries. Hence, improvements in this scenario are becoming more and more needed.

7.2 Short utterances as occluded utterances

The short utterance problem is widely known within the speaker recognition community [Poddar et al., 2017]. The evaluation of trials by means of short utterances involves a severe degradation of performance. However, there is no standard definition of short utterance in the literature. While some works have reported losses of performance with audios containing less than 30 seconds of speech, a more severe degradation is obtained considering shorter utterances (less than 10 seconds) [Mandasari et al., 2011, Kanagasundaram et al., 2011]. This short utterance problem has also been analyzed in the Speaker Recognition Evaluations (SRE), proposed by NIST. Despite traditionally considering utterances with more than 2 minutes of audio, some of the evaluations [NIST, 2008, NIST, 2010] also include a condition in which utterances contain less than 10 seconds.

This loss of performance is a consequence of a higher intra-speaker variability in the estimations with short utterances. In the literature multiple contributions have been proposed to the different steps of the speaker verification pipeline, aiming to reduce the undesired variability. The feature extraction step has been studied in different ways, attempting to provide an alternative to traditional MFCCs. In [Li et al., 2015] a multi resolution time-frequency feature extraction was proposed, carrying out a multi-scaled Discrete Cosine Transform (DCT) on the spectrogram, combining the information afterwards. Alternative works like [Alam et al., 2015] fuse different features based on the amplitude and phase of the spectrum. Other contributions are focused on the modelling stage. Factor Analysis approaches were considered in [Vogt et al., 2008] to develop subspace models to better work with the short utterances. When considering i-vector representations, compensation techniques such as [Kanagasundaram et al., 2013, Kanagasundaram et al., 2014] project the obtained representations into subspaces with low variability due to short utterances. In [Sarkar et al., 2012] it is shown that systems trained on short utterances should compensate the uncertainty due to limited audio, improving the evaluation of short audios. However, when systems must deal with audios with unrestricted length, systems should be trained on long utterances for a better performance. The balance of the Baum Welch statistics, required for the extraction of i-vectors, is also proposed in [Hautamäki et al., 2013]. Besides, DNNs have also mapped short-utterance i-vectors with respect to their long-utterance counterparts [Guo et al., 2017]. Other contributions

have also worked on the backend, specially PLDA. Another technique, originally proposed in [Cumani et al., 2013b, Kenny et al., 2013] and analyzed in Chapter 5, makes the PLDA model include an extra term to compensate the uncertainty of the i-vector, which depends on the utterance length. Finally, other strategies compensate the obtained score according to reliability metrics of the involved utterances [Hasan et al., 2013, Mandasari et al., 2013], specially its duration. This idea is extended in [Viñals et al., 2018b], where the Quality Measure Function (QMF) term studies the interaction between enrollment and test utterances. In [Vogt et al., 2010] intervals of confidence are estimated, leading towards considerable accuracy.

Some works such as [Ajili et al., 2016] have studied the impact of the different phonetic content in the embedding representations. According to their results, vowels and nasal phonemes are helpful for discrimination matters. By contrast, other types of phonemes, such as fricatives or plosives, can be misleading during evaluation. Our hypothesis of work applies this idea of phonemes to short utterances. The presence of certain acoustic units boosts the performance of speaker recognition systems. However, these boosting phonemes must be in both enroll and test utterances to be effective. This match in the phonetic information goes beyond the presence of certain phonemes, also requiring a match in the phonetic distribution along the utterance.

In order to explain our perspective let's make an analogy of the short utterance problem with a similar problem, face recognition with occlusions. In the best scenario, both problems contain all possible information. Working with faces we have a complete view of the person of interest, including all the face elements (two eyes, the nose, the mouth, etc.). In speaker recognition we have complete information in an utterance that contains traces for any possible phoneme and its coarticulation. As long as the utterance gets longer and longer the complete information condition is more likely to be achieved. In this scenario performance has improved more and more as long as technologies have evolved.

Now we focus on short utterances. These contain much less speech, even less than a second. A simple "Yes/No" reply to a question can constitute an utterance. Hence, short utterances are very likely to lack of phonemes. In face recognition the equivalent scenario is the recognition of partial information, where some parts such as the mouth and nose are not visible. In both cases the missing information exists, but it is unavailable. Faces always have a mouth and a nose although sometimes they can be occluded, e.g. by a scarf. Regarding speaker characterization, speakers pronounce all the phonemes of a language while talking, although few of them can be missing in a specific utterance.

In our hypothesis we also consider the influence of proportion. According to our analogy of face recognition, faces present a fixed set of elements (ears, nose, mouth, etc.) with a constrained size, and located in the face in specific areas. These restrictions are always the same,

regardless of the person nor any occlusion. In speaker characterization the situation is slightly different. When utterances get long enough the language imposes restrictions in the phoneme distribution. These restrictions lead to a reference phoneme distribution. The longer the utterance the more its phoneme distribution tends to the reference distribution. However, short utterances contain a much shorter message, and thus its phoneme distribution can be severely distorted. In this distortion we must take into account both the missing phonemes and those present but conditioned to the message in the utterance. This distortion may lead to utterances from the same speaker with different dominant phonemes, hence complicating the evaluation.

Consequently, the short utterance problem can be interpreted as an occlusion from a complete information scenario. This occlusion may be complete, where long utterances lack from certain phonemes, or partial, in which utterances have their phonemes seen in very different proportions with respect to their counterparts. The available information about the occlusion is important to be aware of. During evaluation we compare how the two speakers pronounce all the phonemes, available or not, so unbalanced information can lead to an unfair comparison.

7.3 Formulation of the embedding extraction with short utterances

Current state-of-the-art speaker verification, as described in Section 2.5, relies on the pipeline embedding-backend. Utterances are first converted into compact representations, the embeddings, which feed the decision backend to obtain the score. Among all available representations, two of the most popular ones are i-vectors and x-vectors. Both have been widely tested in speaker verification obtaining great results. First, we will try to understand how we store the speaker information in these embeddings and then study its drawbacks for short utterances.

7.3.1 General case

The method to compact a variable length utterance into a fixed-length representation is similar for most embedding extraction techniques. Given the utterance O, an ordered set of N acoustic features $O = \{o_1, ..., o_n, ..., o_N\}$, we transform them by function $F(\cdot)$, obtaining the ordered sequence $F(O) = \{f_1, ..., f_n, ..., f_N\}$. This function maps the original feature vector o_n into the speaker characteristics subspace as the projections f_n. Depending on the embedding, projection f_n involves the transformation of the feature vector o_n as well as a small context around (approximately 0.15 seconds). By means of this mapping we attempt to highlight the speaker particularities in the features applying linear (e.g. i-vectors) or non-linear transformations (as

in DNNs). The function $F(\cdot)$ is learnt from a large data pool by data analysis, e.g. by Maximum Likelihood algorithms for i-vectors or Back-Propagation [Rumelhart et al., 1986] with DNNs. Due to the fact that each one of these projections f_n only covers a small period of time, they only have information about few acoustic units. The complete characterization of a speaker requires the study of his/her particularities for all the phonemes. These acoustic units are widespread along the utterance, thus we must combine the effect of all these projections f_n. The usual method to combine the projections is its temporal average. The result is the compact representation $G(O)$, defined as:

$$G(O) = \frac{1}{N}\sum_{n=1}^{N} f_n \tag{7.1}$$

This embedding $G(O)$ keeps track of the phonetic content in the utterance O. However, we can also treat each acoustic unit independently. Many state-of-the-art embeddings, such as i-vectors, can be interpreted as the sum of C representations $G_c(O)$, one per acoustic unit, each one estimated according to N_c projections f_n. According to this reasoning we can express the embedding as:

$$G(O) = \sum_{c=1}^{C} \boldsymbol{\alpha}_c G_c(O) \tag{7.2}$$

The obtained expression describes embeddings as a weighed sum of C estimations $G_c(O)$, each one representing the estimated particularities of the speaker in a single acoustic unit. $G_c(O)$ can also be interpreted as the resulting embedding only taking into account the data related to the phoneme c. All the contributions are weighted by the term $\boldsymbol{\alpha}_c$, the proportion of this acoustic unit in the utterance.

Therefore, embeddings are conditioned to two main parts: On the one hand the stability of the distribution of weights $\boldsymbol{\alpha} = \{\boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_c, ..., \boldsymbol{\alpha}_C\}$. On the other hand the estimations $G_c(O)$, the particularities per phoneme. Both benefit from large utterances. Every language has its own reference phonetic distribution. Hence the longer the utterance the more its phonetic distribution becomes like this reference. Concerning the estimations $G_c(O)$, the more available data, the less uncertain is the estimation.

The average stage is the last step in which we keep track of the phoneme distribution. As a consequence, we cannot distinguish between speaker and phonetic variability afterwards. Further steps in the embedding post-processing or the backend may transform the embedding, but all phonemes are equally treated.

7.3.2 i-vector embeddings

The previously described formulation also matches with the traditional i-vectors. The i-vector modeling paradigm, already described in Section 2.5, explains the utterance O as the result of sampling from a Gaussian Mixture Model (GMM), specific for the utterance with parameters λ_O. This model λ_O is the result of the adaptation from a Universal Background Model (UBM), a large GMM that reflects all possible acoustic conditions. This adaptation process is restricted to only the UBM Gaussian means. Besides, the shift of the GMM Gaussians is tied, and explained by means of a hidden variable $\mathbf{w}_O$, located in the Total Variability subspace, described by matrix $\mathbf{T}$. Mathematically:

$$\boldsymbol{\mu}_O = \boldsymbol{\mu}_{\text{UBM}} + \mathbf{T}\mathbf{w}_O \tag{7.3}$$

where $\boldsymbol{\mu}_O$ represents the supervector mean, the concatenation of the GMM component means, from the target λ_O. $\boldsymbol{\mu}_{\text{UBM}}$ is the supervector mean from the Universal Background Model (UBM), the reference model representing the average behaviour. $\mathbf{w}_O$ is the latent variable for the utterance O, with a standard normal prior distribution and $\mathbf{T}$ is a low rank matrix defining the total variability subspace.

The i-vector estimation looks for the best value for the latent variable $\mathbf{w}_O$ so as to explain the given utterance by means of the adapted model. For this purpose, we estimate the posterior distribution of the latent variable $\mathbf{w}_O$ given the utterance O. The i-vector representation $\overline{\mathbf{w}}$ corresponds to the mean of this posterior distribution. Defined in [Dehak et al., 2011], the i-vector is formulated as:

$$\overline{\mathbf{w}} = \left(\sum_{c=1}^{C} \mathbf{T}_c^T \Sigma_c^{-1} N_c(O) \mathbf{T}_c + \mathbf{I}\right)^{-1} \sum_{c=1}^{C} \mathbf{T}_c^T \Sigma_c^{-1} \tilde{\mathbf{F}}_c(O) \tag{7.4}$$

$$= \frac{1}{N(O)} \left(\sum_{c=1}^{C} \mathbf{T}_c^T \Sigma_c^{-1} \frac{N_c(O)}{N(O)} \mathbf{T}_c + \frac{1}{N(O)} \mathbf{I}\right)^{-1} \sum_{c=1}^{C} \mathbf{T}_c^T \Sigma_c^{-1} N_c(O) \overline{\tilde{\mathbf{F}}}_c(O) \tag{7.5}$$

$$= \left(\sum_{c=1}^{C} \mathbf{T}_c^T \Sigma_c^{-1} \boldsymbol{\alpha}_c \mathbf{T}_c + \frac{1}{N(O)} \mathbf{I}\right)^{-1} \sum_{c=1}^{C} \boldsymbol{\alpha}_c \mathbf{T}_c^T \Sigma_c^{-1} \overline{\tilde{\mathbf{F}}}_c(O) \tag{7.6}$$

$$= \Psi^{-1}(O, \boldsymbol{\alpha}) \sum_{c=1}^{C} \boldsymbol{\alpha}_c \Gamma_c(O) = \sum_{c=1}^{C} \boldsymbol{\alpha}_c \Psi^{-1}(O, \boldsymbol{\alpha}) \Gamma_c(O) = \sum_{c=1}^{C} \boldsymbol{\alpha}_c G_c(O) \tag{7.7}$$

where $\mathbf{T}_c$ represents the portion of the matrix $\mathbf{T}$ affecting the cth component of the UBM. Σ_c symbolizes the covariance matrix for the cth component of the UBM. $N_c(O)$ and $\tilde{\mathbf{F}}_c(O)$ are the

zeroth and centered first order Baum Welch statistics for utterance O. These statistics represent the number of samples from component c and the accumulated deviation with respect to the mean of the same component respectively. $N(O)$ symbolizes the total number of frames in the utterance O. Finally, the term $\overline{\tilde{\mathbf{F}}}_c(O)$ is the average deviation per sample of the utterance for the component c of the UBM.

The formulation of i-vectors offers special characteristics. First, the value of C, the number of traced acoustic units to discriminate, is fixed in the UBM. Its value is equal to the number of Gaussian components in the UBM. Therefore, $G_c(O)$ represents the contribution per sample to the i-vector from component c, and the weight $\boldsymbol{\alpha}_c$ is the proportion of frames assumed to be sampled from same cth component. Furthermore, i-vectors have no speaker awareness in their formulation. They simply store the variations in the acoustic units within an embedding. These deviations from the average behaviour, properly treated by the backend, are responsible for the performance in speaker identification systems.

7.3.3 Short utterances

Now we consider the short utterance scenario. According to the previous analysis, embeddings work well if the distribution of acoustic units $\boldsymbol{\alpha}$ is similar to the reference distribution and the particular contributions $G_c(O)$ are estimated with low uncertainty. These two requirements are reassured as long as the utterance contains more and more data. Concerning short utterances, their low amount of data makes them likely to have their distribution of acoustic units $\boldsymbol{\alpha}$ far from their reference. For the same reason short utterances may also suffer from large uncertainty in their phoneme estimations $G_c(O)$. Hence, degradation in short utterances can be explained by the following reasons:

- **Errors in the contribution of phonemes**. Some contributions $G_c(O)$ were estimated with very little information. Then the uncertainty of their estimation increases. Multiple values within this uncertainty range as $G_c(O)'$ can be estimated instead, committing the error $E = G_c(O)' - G_c(O)$.

- **Mismatch in the phoneme distribution**. The distribution of the weights $\boldsymbol{\alpha}$ does not match the reference $\overline{\boldsymbol{\alpha}}$, defined by language characteristics. This degradation causes the error $E = \sum_{c=1}^{C}(\boldsymbol{\alpha}_c - \overline{\boldsymbol{\alpha}}_c)G_c(O)$. The extreme case happens when some acoustic units are not present in the utterance, i.e. they are missing. In this situation their weight $\boldsymbol{\alpha}_c$ are equal to zero, also forcing the missing estimations $G_c(O)$ to be set to zero, as if they were occluded. The degradation due to the mismatch in the phoneme distribution is compatible with the errors in the contribution of phonemes.

Traditionally errors have been attributed to the contributions per acoustic unit. This is specially true when traditional embeddings, e.g. i-vectors, include an uncertainty term in its calculations. For this reason, this sort of error was the first attempted to deal with, e.g. [Kenny et al., 2013]. However, to the best of our knowledge no previous work has covered the degradation due to the phoneme distribution, which can cause similar levels of degradation.

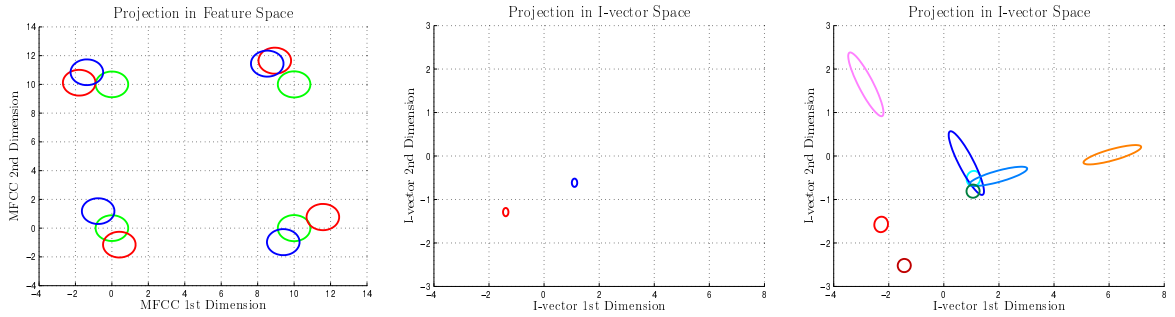
7.4 Effects of the short utterances in i-vectors

The phonetic distribution in an utterance has important implications during the embedding extraction. Embedding shifts due to incorrect contributions $G_c(O)$ are complementary to those created by the mismatch in the phonetic distribution. In this section we illustrate their impact with i-vectors. This choice of well-known embeddings makes the study of both problems more illustrative in a simple way.

For this purpose, we propose a small dimension i-vector experiment to test the effects of short utterances in some artificial controlled data. Given an evaluation UBM i-vector pipeline, we compare the i-vectors obtained from an original utterance and those obtained from the same utterance after undergoing controlled short-utterance modifications. These modifications affect both the acoustic unit distribution $\boldsymbol{\alpha}$ and their contributions $G_c(O)$. We make use of the following experimental setup: We first sample a large artificial data pool from a UBM i-vector pipeline. This data pool consists of more than ten thousand independent utterances, with one hundred two-dimension samples each. The UBM is a 4-Gaussian GMM whose components are located in $(0, 0), (0, 10), (10, 0)$ and $(10, 10)$, all of them with the identity matrix as covariance. The generative i-vector extractor has a 3-dimension hidden variable subspace. With ten thousand of these utterances we train our evaluation pipeline, an alternative UBM i-vector system. For simplicity we share the generative UBM. Regarding the i-vector extractor, we train a model with only a two-dimension latent subspace. This dimension reduction between generation and evaluation has been considered to imitate real life, where the generation of data is a too complex process that we only can approximate.

From the remaining data pool we choose two extra utterances, unseen during the model training, for evaluation purposes. Because these two utterances are independent, we assume them to represent two different speakers. In Fig. 7.1 we represent them, red and blue respectively. The representation includes three parts: In the first part we show the original feature domain, i.e. the utterance set of feature vectors. Each ellipse in the figure represents the distribution of each Gaussian in their GMMs. The image also includes in green the representation of the UBM model. The second part in Fig. 7.1 represents the same red and blue utterances in

the latent space by means of the posterior distribution of the latent variable $\mathbf{w}$. The third part of Fig. 7.1 illustrates the location of the particular estimations per component $G_c(O)$ for the two utterances in the latent space. Reddish estimations correspond to the red speaker while bluish ellipses represent the phonemes for the blue speaker.



a) Data domain b) i-vector domain c) Components in Iv. domain

Figure 7.1: Scenario of interest. a) Utterances red and blue in the feature domain, with the UBM components in green. b) Utterances red and blue in the i-vector domain. c) Projections of the GMM components in the i-vector domain for utterances red (reddish ellipses) and blue (bluish ellipses).

Following the described setup we can carry out an analysis of degradation in short utterances. First, we illustrate the phoneme dependent estimation error due to limited data. For this reason we estimate the posterior distribution of the embeddings for multiple utterances only differing the number of samples. The distribution of phonemes $\boldsymbol{\alpha}$ remains unaltered. Theoretically, the embeddings should not suffer any bias, but its uncertainty should get larger as long as the utterances contain less data. In Fig. 7.2 we compare the original utterances to those obtained with one fifth of the data and one tenth of the data.

Fig. 7.2 illustrates the posterior distribution of the latent variable for the short utterances (dashed-line red and blue ellipses) as well as the original utterances (red and blue ellipses with continuous line respectively). The location of the ellipse represents the mean of the posterior distribution while its contour the uncertainty. As expected, the original reference utterance and their shorter versions present very reduced shifts among themselves, with almost concentric ellipses. While the blue speaker suffers almost no degradation, the red speaker biases are more noticeable. Besides, the illustration shows that the less data in the utterance, the bigger the uncertainty of the estimation.

Now we study the impact of the distribution of acoustic units $\boldsymbol{\alpha}$ on the embedding. In the reference utterances this distribution was uniform, this is, 25% of the samples came from each component. We now modify this distribution for both utterances, red and blue. In Fig. 7.3 we show the posterior distributions of the original utterances (red and blue ellipses with continuous

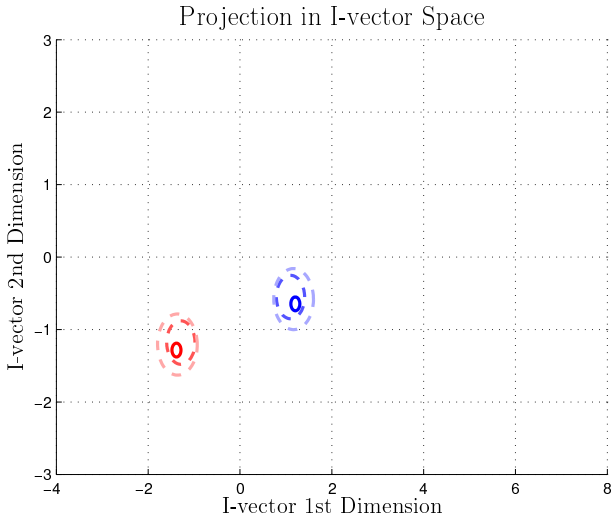


Figure 7.2: Comparison of posterior distribution of the i-vectors with reference phoneme distribution. Continuous line ellipse represents the original utterance while dashed-lined ellipses illustrate utterances with the limited data.

line) as well as the altered short utterances (dashed-line red and blue ellipses). In the illustrated example half of the feature vectors are sampled from a single component of the GMM while the remaining data is evenly sampled along the other components. We have studied the effect with the four components in the GMM.

Illustrated results in Fig. 7.3 reveal the relevance of the distribution of phonemes $\boldsymbol{\alpha}$ for its proper modelling. The modification of the distribution of weights makes the red speaker to offer four different representations of the same embedding. Besides, these representations are not overlapped among themselves, beyond the uncertainty region from the original utterance. Therefore, these alternative embeddings are likely to fail. Nevertheless, not all speakers behave equally. Whilst red speaker is degraded, our blue speaker has suffered the same alterations without any visible shift on his/her embeddings.

The scenario with a distorted phoneme distribution can be taken to the limit. In this situation some components do not contribute to the final embedding. This scenario is the most adverse, significantly modifying the distribution of patterns $\boldsymbol{\alpha}$ and some estimations per phoneme $G_c(O)$ being set to zero. In this experiment we have disturbed the distribution of acoustic units $\boldsymbol{\alpha}$ forcing two of the components to zero. In Fig. 7.4 we illustrate the six possible scenarios in terms of the non-contributing components. The results are shown for the two test speakers red and blue, with continuous line ellipses for the reference utterances and dashed-line ellipses for their altered versions. According to the representations shown in Fig. 7.4, embeddings from utterances with missing components experiment large biases with respect to the reference

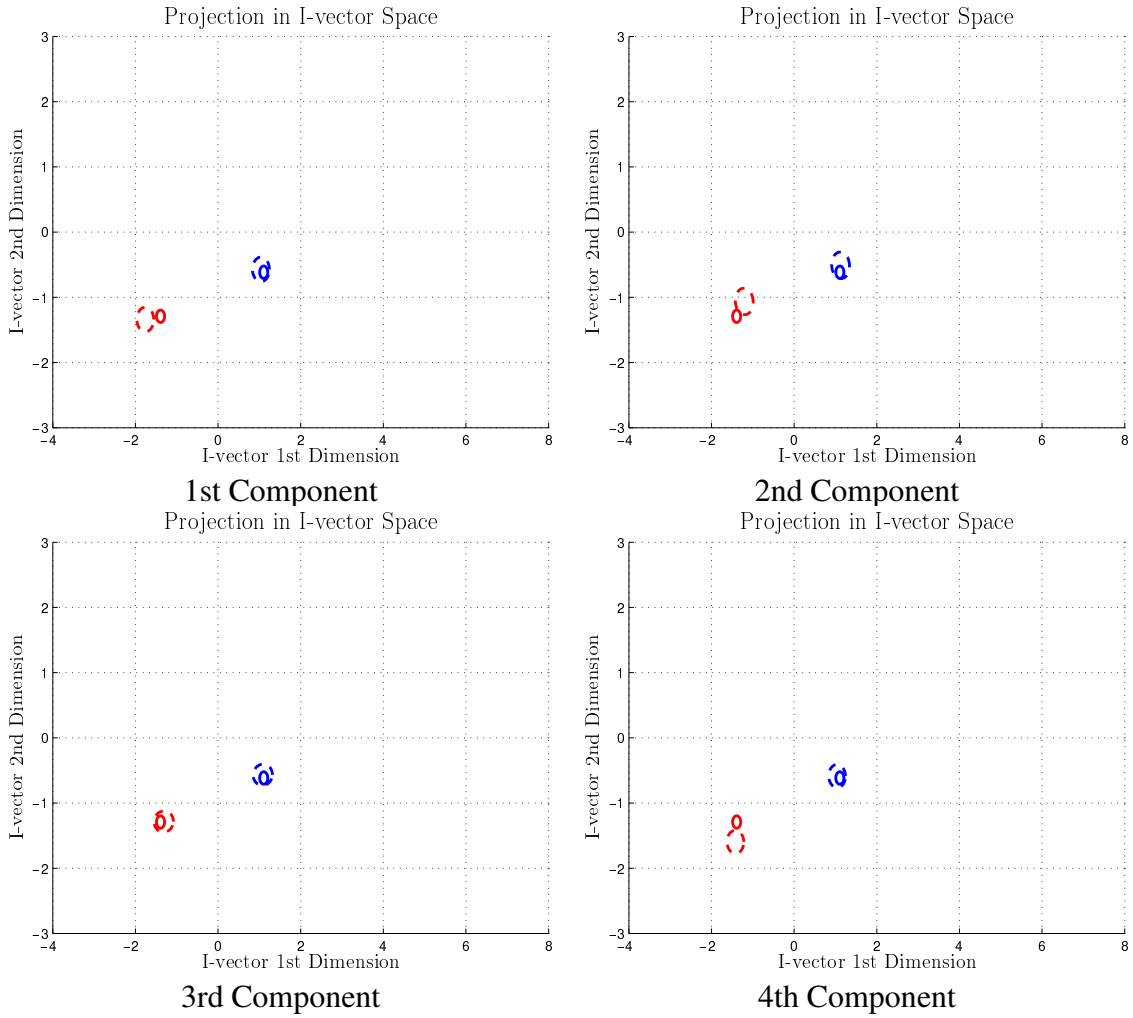


Figure 7.3: Comparison of posterior distribution of i-vectors with modifications in the phoneme distribution α

embeddings. These shifts are more significant than those previously seen with less extreme distortions in the phoneme distribution α. Some of the hypothesized embeddings are far beyond the uncertainty from the original utterance. The biases suffered by the utterances are not the same for both speakers. Again, the blue speaker suffers no relevant degradation. This behaviour fits in our hypothesis because the missing components scenario is the limit case of phoneme distribution degradation.

In all our experiments the red speaker has suffered from strong degradations while the blue speaker has remained almost unaltered. This different behaviour is a consequence of the locations of the phonetic estimations $G_c(O)$ for each speaker. On the one hand, as shown in Fig. 7.1, our blue speaker has its components very close to each other, providing robustness against distribution modifications. On the other hand our red speaker has its components much further

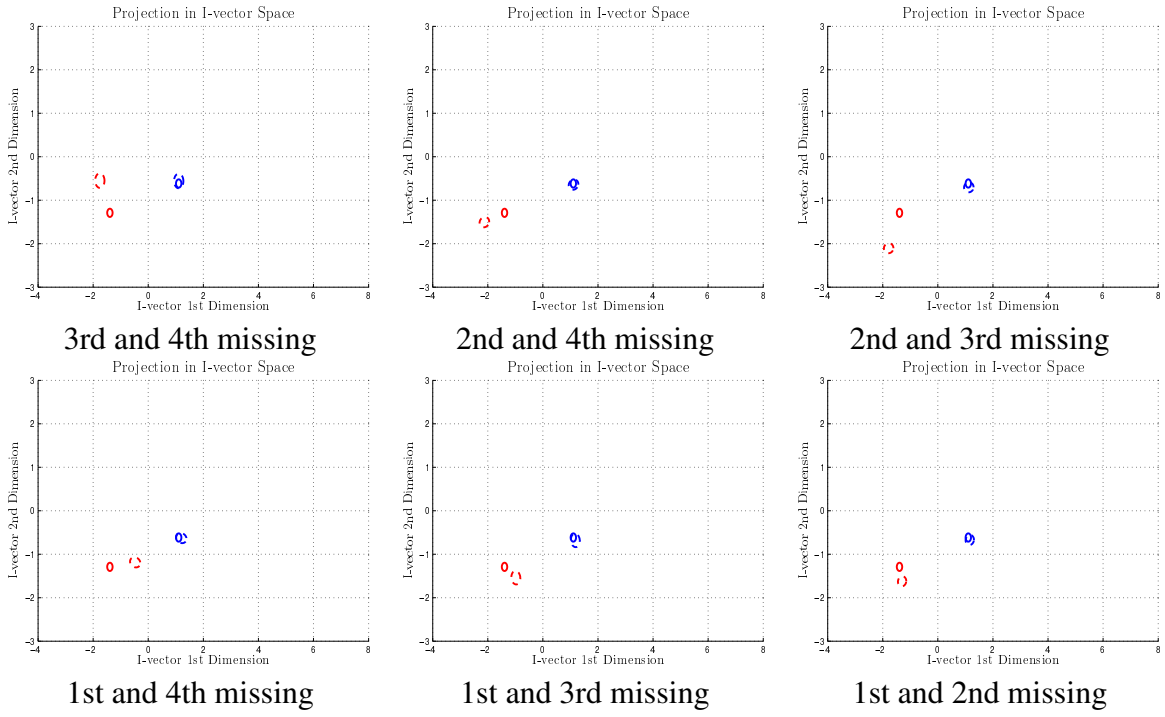


Figure 7.4: Comparison of posterior distribution of i-vectors when two phonemes are not contributing and $\boldsymbol{\alpha}_c = 0$

from each other. Therefore, any alteration of the distribution implies a much more significant degradation in the location of the red speaker. In consequence, some speakers will be more robust to short utterances modifications than others.

7.5 Experiments & Results

Our hypothesis is that short utterances work well in evaluation if both enrollment and test contain similar phonetic content, being degraded otherwise. According to our previous analysis with artificial data, embeddings from short utterances can suffer from biases due to a mismatch in the distribution of acoustic units $\boldsymbol{\alpha}$ and the effect of missing components $G_c(O)$. Therefore, evaluation of trials should behave better if both enrollment and test embeddings were similarly altered.

7.5.1 Experimental setup

Our experimental scenario works around the NIST SRE evaluations, specifically the SRE10 "coreext-coreext det5 female" experiment, already considered in Section 5.2.1. In this work we restrict our efforts to i-vectors. This choice was taken for illustrative purposes. Therefore, 20

MFCC feature vectors, with first and second order derivatives and Short Time Gaussianization [Pelecanos and Sridharan, 2001] are applied. Utterances are then represented by a gender dependent 2048-Gaussian UBM trained with excerpts from SRE 04,05,06 and 08. Based on this UBM a gender dependent 400-dimension T matrix is trained, also using excerpts from SRE 04, 05, 06 and 08. The obtained embeddings, in this case i-vectors, are centered, whitened and length-normalized [Garcia-Romero and Espy-Wilson, 2011]. The back-end consists of a 400 dimension Simplified PLDA. No score normalization nor calibration are applied, so results are measured in terms of Equal Error Rate (EER) and minDCF. For evaluation purposes we make use of the original evaluation operation point.

In this work we also need to measure the relative differences in the phoneme distribution between enrollment and test. Hence, we must define a metric to measure how close these two utterances are from each other. In this work we have opted for the KL2 distance (Section 2.4.1.2). In i-vectors the phoneme distribution $\boldsymbol{\alpha}$ matches the responsibility distribution. Consequently, our KL2 metric is evaluated between the responsibility distribution of enrollment and test. This distribution can be obtained from the zeroth order Baum-Welch statistic.

7.5.2 Baseline

Our first experiment sets a benchmark based on the SRE10 "coreext-coreext det5 female" experiment. This experiment only includes utterances with approximately 300 seconds of audio. Hence, in order to illustrate the degradation of short utterances we have considered two datasets obtained from the same utterances.

- **Long**. The original utterances provided by the organizers for the evaluation, with approximately 5 minutes of audio per utterance. These utterances play the role of long reference utterances.

- **Short Random**. An alternative version of the original SRE10 dataset with restricted information. For this purpose, we will follow the procedure described in Section 5.2.1 to craft short utterances. Each utterance of the original dataset is chopped restricting its audio speech to be in the range 3-60 seconds. The chop marks, starting point and initial position were randomly chosen. Utterance chopping was done after VAD. These utterances can suffer from degradation due to errors in the phoneme estimations $G_c(O)$ and mismatch in the phoneme distribution $\boldsymbol{\alpha}$.

Thanks to these two datasets we have available a version of the utterances with full information and a version with partial knowledge. Now we must define the scenarios for the evaluation,

Table 7.1: Results, EER(%) and minDCF of SRE10 "coreext-coreext det5 female" experiment with the three scenarios of interest: Long-Long, Long-Short and Short-Short

Scenario	EER(%)	MinDCF
Long-Long	3.25	0.16
Long-Short Random	**5.67**	**0.27**
Short-Short Random	8.57	0.40

assigning the roles of enrollment and test. We are interested in three particular scenarios:

- **Long-Long**. The official NIST SRE10 experiment. The Long dataset plays both roles, enrollment and test, in each trial. This experiment represents the case in which we have complete information for both speakers.

- **Long-Short**. In this scenario the Long dataset also plays the role of enrollment, while the shortened dataset is used for test. In this scenario we study the scenario where the reference speaker, the enrollment, is perfectly characterized while the candidate (the test) speaker is unreliably represented.

- **Short-Short**. The Short Random dataset plays both roles, enrollment and test. This scenario reveals the performance with very limited information.

The three scenarios are evaluated by means of the same trial list, which defines the comparisons to evaluate. The only difference among scenarios is the particular audio within the utterance to model the speakers. The results with these three configurations can be seen in Table 7.1. These results confirm that short utterances degrade performance. Besides, as long as more and more data are represented by means of short utterances, we suffer more degradation. This degradation affects both evaluation metrics EER and minDCF.

7.5.3 Reduction of the mismatch in α: Phonetic balance

In our previous analysis we hypothesized two main sources of degradation in short utterances: The one due the uncertainty in the phoneme estimations $G_c(O)$ and another term caused by mismatches in the phoneme distribution $\boldsymbol{\alpha}$. In order to test our hypothesis we are going to minimize the errors due to phoneme distribution $\boldsymbol{\alpha}$. To do so we have prepared an extra dataset, named as **Short Balanced**. This dataset is also obtained from the original data released by

Table 7.2: Comparison of results, EER(%) and minDCF, between Short-Random and Short Balanced dataset in SRE10 "coreext-coreext det5 female" for scenarios Long-Short and Short-Short.

Scenario	EER(%)	MinDCF
Long-Short Random	5.67	0.27
Long-Short PHN Balanced	**3.62**	**0.19**
Short-Short Random	8.57	0.40
Short-Short PHN Balanced	**4.11**	**0.20**

the organization. From each original utterance we obtain phoneme labels, one per input feature vector. These phoneme labels in this experiment were obtained by automatic means, i.e., a DNN phoneme classifier [Viñals et al., 2019d] consisting of a Wide Residual Network [Zagoruyko and Komodakis, 2016] with four blocks. Only 39 phoneme labels were considered, i.e., each phoneme label includes all its associated coarticulation. Experiments carried out in TIMIT [Garofolo et al., 1993] show error rates around 15 % in the classification task.

The phoneme labels in an utterance determine its phoneme distribution. This distribution, obtained from the long utterance, must be maintained in the new short utterance despite its lower length. Therefore, given the length of the new short utterance we can determine the newer number of samples per phoneme. Then we randomly choose this number among the all the samples with a certain phoneme, repeating the process for all phonemes. Frames must be considered speech by our VAD to be candidate for the new utterances. For comparison reasons each Short-Balanced utterance contains as many samples as in the Short Random counterpart.

The comparison between both types of short utterances is shown in Table 7.2. The comparison includes the results in the Long-Short and Short-Short scenarios. This information is complemented with the DET curves in Fig. 7.5, where we also include the Long-Long scenario for comparison reasons.

According to the shown results, the new Short Balanced dataset is able to behave much better than the Short Random dataset, despite containing both datasets the same amount of speech. This is because the phonetic balance with respect to the original utterance also reduces the distance between enrollment and test phoneme distributions $\boldsymbol{\alpha}$. Therefore, we get rid of this source of error, only remaining those errors due to the phoneme estimations $G_c(O)$. We have also realized that the degradation due to the phonetic distribution mismatch is much more relevant than those related with the uncertainty of the estimations $G_c(O)$.

In order to obtain a better understanding we analyze the already obtained results in terms of the type of trial: target and non-target. This study compares the KL2 distance between

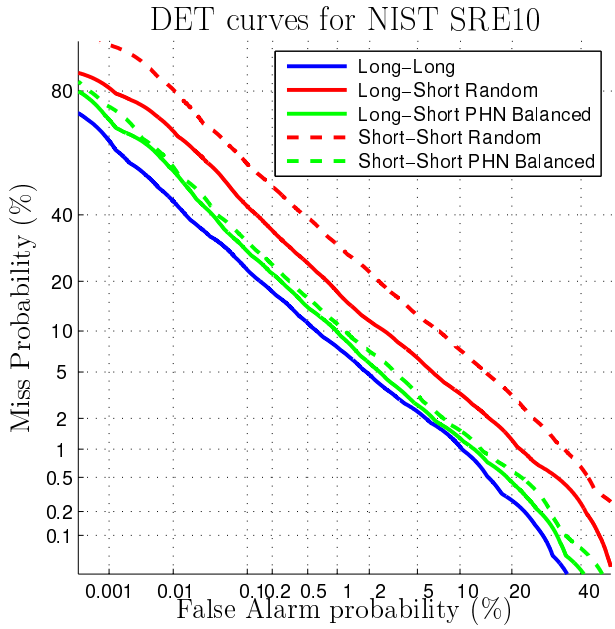


Figure 7.5: DET curves for the scenarios Long-Long (blue), Long-Short and Short-Short Random (red continuous and dashed line respective), Long-Short and Short-Short Balanced (green continuous and dashed line respective) for SRE10 "coreext-coreext det5 female" experiment.

enrollment and test with respect to the probability of error in each population. In this work the KL2 distance measures how the responsibility distributions for enrollment and test utterances match each other. The obtained results are shown in Table 7.3. This analysis is performed for the experiments Long-Long, Short-Short Random and Short-Short Balanced. Besides, we study the impact of the KL2 to the classification error in target and non-target trials, i.e., the Miss and False Alarm error terms. The decision threshold is set up according to the operating point defined by the evaluation.

Results in Table 7.3 illustrate many details. First, the KL2 distance increases for both target

Table 7.3: KL distance and Error (%) for both target and non-target trials in experiments Long-Long, Short Short Random and Short-Short Balanced. Error estimated at NIST operating point.

Experiment	KL2 Distance (nats)		Population Error (%)	
	Target	Non-Target	Target	Non-Target
Long-Long	1.06	1.74	28.43	0.06
Short-Short Random	3.62	4.61	80.40	**0.01**
Short-Short Balanced	2.55	3.47	**40.79**	0.03

and non-target trials as long as we move from the Long-Long experiment to the Short-Short Balanced and finally the Short-Short Random experiment. Besides, this KL2 distance is always higher in the non-target trials population than in target trials. Moreover, regardless of the utterance length or content, evaluation errors are mainly caused by the misclassification of target trials. We also see some correlation between the relative KL2 distances and the errors. In target trials the lower the distance, the lower the error in the target population. These results also illustrate that our Short-Balanced dataset obtains its improvement mainly from the target trials, halving their error. With respect to the non-target trials, we see a negative correlation, with an error term deceasing as long as the KL2 metric increases.

Finally, combining the information of Table 7.2 and Table 7.3 we realize that the relationship between the relative KL2 distance and the error metrics EER and minDCF is not linear. We carried out a linear regression of the evaluation results (EER and minDCF) in terms of the KL2 distance. This regression was estimated in terms of the obtained results for the Long-Long and Short-Short Random experiments. When we infer the results for the Short-Short Balanced experiment according to its KL2 distance, those are significantly worse than the really obtained ones. According to this regression this experiment should have obtained 6.33% EER and 0.30 minDCF, far higher than the obtained values. Therefore, the distance/EER and distance/minDCF relationships should be steeper with higher KL2 values and more even with the lower distances.

7.5.4 Enrollment-test distance vs log-likelihood ratio (llr)

Our previous experiments reveal the relationship between the relative distance in terms of phonetic content between enrollment and test utterances and the performance of these trials. Thus, we explore the impact of this distance in the performance.

For this analysis we opt for the Short-Short scenario. The test role in each evaluation is always played by an utterance from the Short Random dataset. Regarding the enrollment role, we have created the following pool of data, with multiple candidate utterances for each trial:

- The Short Random dataset experiment previously analyzed.
- Three alternative Short Random datasets. We follow the crafting process described in Section 5.2.1 to generate short utterances in the range of 3-60 seconds of speech. Nevertheless, now the short segments are not totally random. They are restricted to differ from the test utterance in the trial a controlled KL2 value. These goal values for the distances are approximately 2, 3 and 4 nats.

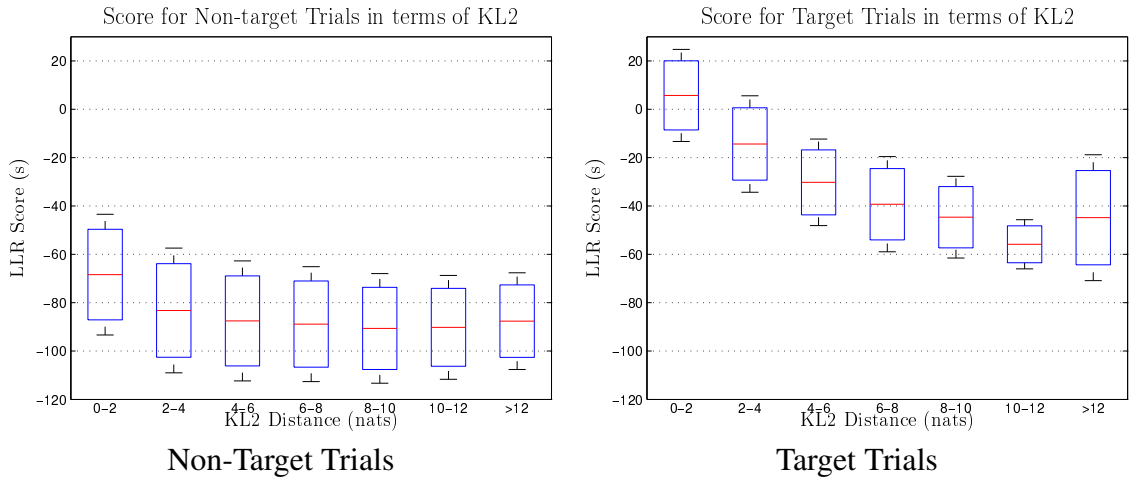
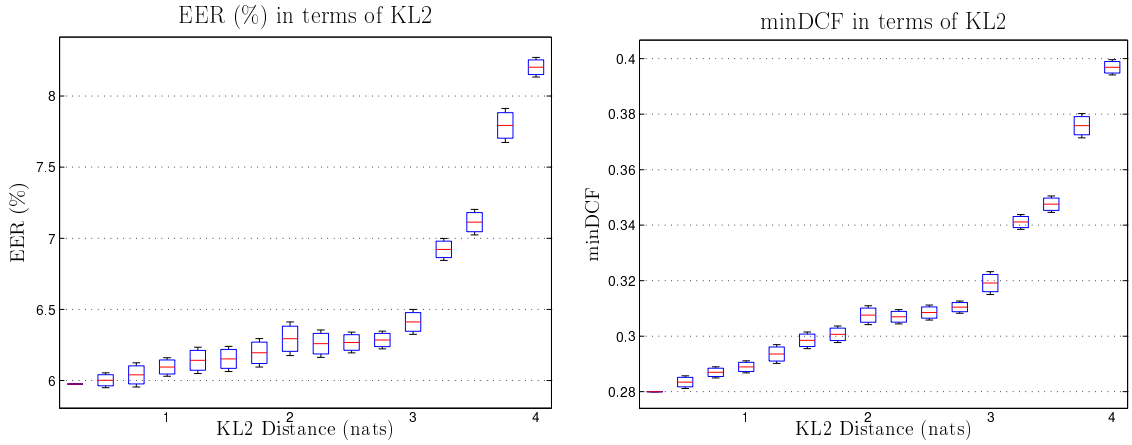


Figure 7.6: Trial score in terms of KL2 distance for the whole data pool. Represented the mean and the mean plus/minus the standard deviation

- Short Equalized dataset. We equalize the original enrollment utterance to obtain a null relative distance between enrollment and test. By doing so we choose from the enrollment only those contents present in the test utterance and in the same proportion. The amount of audio is the same in both enrollment and test utterances.

This large data pool allows the analysis of the relationship between enrollment-test relative distance and score. The results are visible in Fig. 7.6. We present a boxplot of the log-likelihood ratio score in terms of the distance in bins of 2 nats. Three values per bin are shown: the mean of the scores, the mean plus the standard deviation of scores and the mean minus the standard deviation of scores. We have differentiated between non-target trials and target trials for a better understanding.

Fig. 7.6 confirms our previous conclusions. First, the score of target trials is strongly influenced by the relative phoneme distance between enrollment and test. The lower the distance, the higher is the score for the target trials. By contrast, non-target trials are almost insensitive to this distance. Their scores remain steady for almost all the analysis. In conclusion, degradation is mainly caused by target trials, which strongly depend on the KL2 metric. However, Fig. 7.6 indicates something more. Non-target trials keep stable for almost all the distance range except for low values (0-2 nats), where the score increases. The very high phonetic similarity between enrollment and test utterances increases their log-likelihood ratio despite these trials do not contain the same speaker.



a) KL2 vs EER b) KL2 vs minDCF

Figure 7.7: Evaluation metrics, EER (a) and minDCF (b) in terms of the KL2 distance. Represented the mean and mean plus/minus the standard deviation of the metric per bin.

7.5.5 Enrollment-test distance vs performance (EER and minDCF)

Previously we have studied the effect of relative phoneme distances on the score, individually analyzing each trial. Now we study the whole set of trials at once, providing the evaluation metrics, the Equal Error Rate (EER) and the minimum Decision Cost Function (minDCF).

In Fig. 7.7 we analyze the impact of the relative phonetic distance for the two evaluation metrics. For this purpose, we have conducted the SRE10 "coreext-coreext det5 female" experiment, selecting the scores for each trial from the previously described pool of scores. The results show the performance in terms of the average KL2 distance between enrollment and test trials. More than 10000 different score sets were studied.

Fig. 7.7 also confirms our previous conclusions. Previously we inferred a non-linear behaviour of the EER and minDCF with respect to the relative phoneme distance. We realized that low values of phoneme distance should generate low degradations, getting more relevant as long as the KL2 distance increases. Fig. 7.7 shows an elbow shaped relationship with two different behaviours. Below a certain value, in this case approximately 3 nats, both evaluation metrics experiment low degradations (0.5% EER and 0.03 minDCF). However, once the relative phoneme distance exceeds this value, degradation increases rapidly. This elbow shape has great implications. By working within the lowest range of relative phoneme distance (in our case below 3 nats) we can assume a certain reliability in our results.

Table 7.4: Comparison of results, EER(%) and minDCF, for scenarios Long-Short Random and Short-Short with equalized results. SRE10 "coreext-coreext det5 female experiment".

Utterance	EER(%)	MinDCF
Long-Short	**5.67**	**0.27**
Short-Short Equalized	5.98	**0.27**

7.5.6 Long-short vs Equalized Short-Short

Our experiments in the Short-Short scenario have revealed that utterances are better classified as long as the relative phonetic distance between enrollment and test decreases. Nevertheless, further information is available in other scenarios, as in the Long-Short scenario. While the short utterance has limited information in it, maybe missing some phonemes, the long utterance has complete information about all phonemes. Hence, we must check whether this extra information is worthy or not.

For this reason, we compare the originally defined Long-Short experiment with the Short-Short experiment with lowest relative phoneme distance. This Short-Short experiment implies the equalization of the enrollment utterance to match the phonetic content in the test utterance, getting rid of any extra information. In this comparison both experiments share the same test utterances. The results for this experiment are shown in Table 7.4 and DET curves are shown in Fig. 7.8.

The results in both Table 7.4 and Fig. 7.8 show that the extra information has a very small effect in the evaluation task. Nevertheless, despite both experiments have obtained very similar results, the Long-Short original experiment is slightly better. This issue can be partially justified by the range of the relative phoneme distances of the short-short experiment. The equalization imposes the relative distance to be equal to zero. In this range of values the non-target trials experiment an increase of the score, possibly causing the degradation. In order to confirm this explanation we analyze the score distribution for population of target and non-target trials in both experiments. These distributions are represented in Fig. 7.9:

Fig. 7.9 confirms our hypothesis of harmful non-target trials. The scores for the target trials overlap in both scenarios, Long-Short and Short-Short Equalized. By contrast, the score distribution for the non-target trials in the Short-Short equalized experiment is slightly positively biased with respect to the Long-Short counterpart. This extra deviation is responsible for the experimented small degradation of performance.

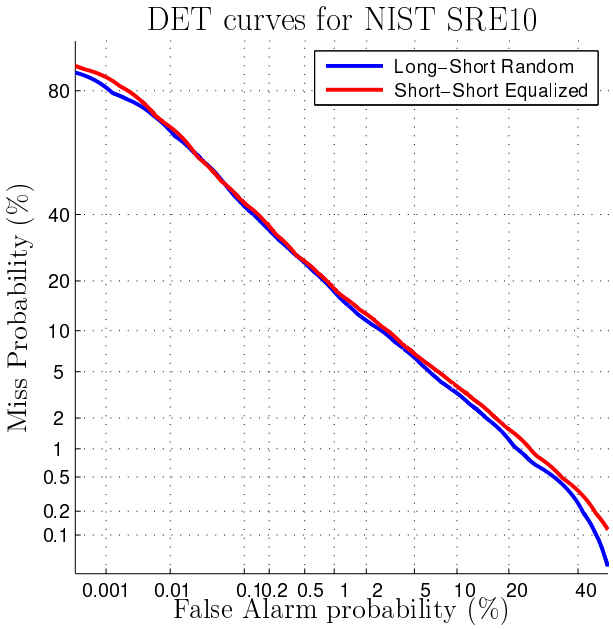


Figure 7.8: DET curves for the scenarios Long-Short Random and Short-Short Equalized in SRE10 "coreext-coreext det5 female".

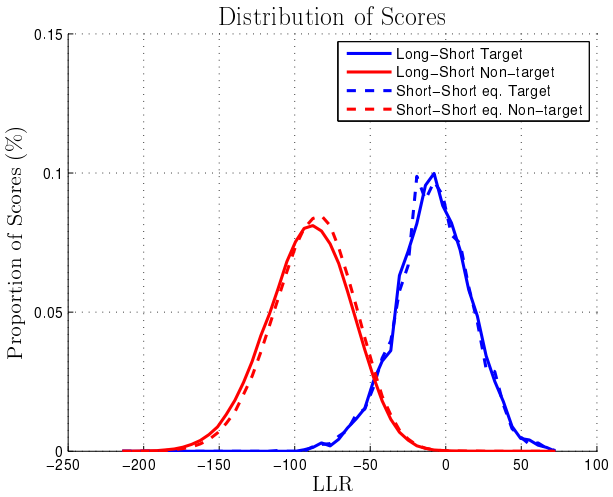


Figure 7.9: Normalized distribution of scores for Target (blue) and Non-target(red) trials of scenarios Long-Short (continuous line) and Short-Short Equalized (dashed line).Experiment carried out with SRE10 "coreext-coreext det5 female".

7.6 Conclusions

In this chapter we have successfully analyzed the problem of short utterances as a problem of unbalanced, even missing, patterns.

We have shown that embeddings can be partially understood as weighted sum of phoneme contributions, each of them illustrating the particularities of the speaker for the acoustic unit. When the weight distributions differ from the expected one as in short utterances, embeddings experiment shifts with respect to their original location. When these shifts are large enough they are not considered intra-speaker variability anymore, and attributed to speaker mismatches. Therefore, these shifts are responsible for the loss of performance.

Our contribution has been focused on the phonetic similarity between enrollment and test utterances. We have proposed the KL2 distance as metric for the relative phonetic distance between enrollment and test. We have also illustrated the dependencies of the score and the evaluation performance (EER, minDCF) of systems in terms of the proposed distance. Moreover, we have realized that this influence is specially noticeable in the target trials, while non-target trials are almost unaffected. Our results also indicate the existence of a range of reliable distance where degradation is bounded. Working beyond this limit makes performance degrade very fast. Furthermore, our experiments indicate that once perfect match of the distributions is achieved, further information in extra components does not provide a significant improvement in performance.

Unfortunately, the phoneme distribution must be complemented with accurate information for all possible phonemes to be improved. Our experiments with very low relative phoneme distances, even zero, behave worse than experiments with complete information but larger enrollment-test distances. This is a consequence of the unseen phonemes, which help with the distance but not with the discrimination of speakers. Further research should be done about this missing information. Moreover, this analysis has been carried out with i-vectors. Therefore it is required experimental confirmation with other embedding representations in the state of the art.

Chapter 8

DNNs embeddings for Diarization

Along the previous chapters we have evaluated a diarization system based on i-vectors, state of the art at the very beginning of this work. However, the state of the art has evolved for the latest years moving towards a new paradigm where the exclusive statistical representation of speakers has been overcome by DNN-based strategies.

Thus, the goal for this chapter is the study of alternative types of embedding with respect to i-vectors. For this purpose, we will explore the new DNN paradigm within speaker recognition, analyzing the advantages and disadvantages for these embeddings as well as their incorporation to broadcast diarization.

8.1 Introduction

Since their appearance in [Dehak et al., 2011] i-vectors have led text-independent speaker recognition state-of-the-art. Their great results in text-independent speaker verification motivated their inclusion in diarization as well [Villalba et al., 2015][Diez et al., 2018]. However, pattern recognition state of the art has evolved moving from traditional statistics based on linear models (GMMs, JFA, i-vectors) to non-linear models, significantly more complex yet much more powerful approaches. Among these options, the Neural Networks (NNs) with several layers, also known as Deep Neural Networks (DNNs) have taken the lead within the speech community.

Neural networks in speech community are not a novelty e.g. [Morgan et al., 1991][Waibel et al., 1989]. However, original networks were specially limited by data availability and hardware computational capabilities. The hardware improvements in the early 2000s as well as the evolution of algorithms allowed the uprising of DNNs, i.e., neural networks of several layers to increase their modeling capabilities. In [Hinton et al., 2012] DNNs were applied to speech technologies, integrating them into state-of-the-art ASR

techniques. This integration was done by substituting a traditionally linear block by its DNN counterpart, more complex but much more accurate, becoming this substitution state of the art. Since then, ASR performance has been improving by the use of more and more DNNs to deal with tasks previously done by linear models, as well as by improving the modeling capabilities of these networks [Graves and Schmidhuber, 2005][Mikolov et al., 2013][Graves et al., 2013].

Moving towards speaker recognition and diarization, the attempts to properly include DNNs were not so straightforward. A relevant difference between speaker recognition and ASR is the amount of information available per label. While ASR neural networks, specially those developed for acoustic modeling, make a decision for a limited amount of audio (around 100 labels per second), speaker recognition community wanted to classify among speakers, i.e. 1 label per utterance, regardless of its length. Consequently, the training for speaker recognition DNNs was not possible at that time. Instead, speaker recognition generated hybrid solutions by assisting the traditional i-vectors with DNN information. These systems are known as hybrid i-vectors. In order to include these neural networks into i-vectors authors realized that this statistical model only depends on the Baum-Welch statistics, obtained in terms of a UBM. Therefore, DNNs could be applied by influencing these statistics, either in the responsibilities [Lei et al., 2014] as well as at the feature level with the Bottleneck features [Zhang et al., 2014]. Some works have also combined their performance [Sadjadi et al., 2016].

The impact of DNNs in speaker recognition was direct, reducing the error rates to unprecedent levels. Nevertheless, new advances in neural networks led to the x-vector [Snyder et al., 2016], the first purely DNN characterization tool. This embedding is extracted from the forward propagation of the utterance information along the network, which was previously trained to discriminate speakers in a closed-set setup. Interestingly enough, in contrast to i-vectors and its imposed Gaussian nature, x-vectors do not restrict embedding distribution, letting the network learn it by itself.

In the following sections we will analyze how some of these technologies are suitable for our broadcast diarization purposes.

8.2 Hybrid i-vectors

According to its definition, the i-vector paradigm works under the premise that the total acoustic variability can be modeled by a GMM playing the role of UBM, being individual utterances theoretically sampled from a particular GMM adapted from the average model. Besides, this adaptation has some restrictions imposed in such a way that it only affects the GMM means, which are tiedly shifted, i.e. their shifts are interconnected, in terms of a latent variable lying

within a low dimensional space.

The study carried out in Chapter 7 analyzing the i-vector formulation has revealed that i-vectors can be decomposed as the summation of contributions, each one obeying to a particular pattern. Furthermore, due to its own definition, no speaker knowledge is mapped in i-vectors. Hence, these embeddings simply model the acoustic content along the utterance with respect to an average speaker, represented by the UBM. These variations are then mapped into a low-dimensional representation, the i-vector. It is up to posterior steps, like the backend, the decision of which shifts are more discriminative among speakers.

Before we have mentioned that i-vectors map different patterns. However, GMMs do not require labels to classify the frames, e.g. acoustic units or phonemes. Instead, an unsupervised clustering is done, causing that certain phonemes may be represented along multiple GMM components. Due to our acquired knowledge a significant improvement might be obtained by means of a more restricted component assignment. This alternative alignment could reduce the variability for each acoustic and helping to differentiate the multiple pronunciations of the same sound. Hybrid i-vectors are the DNN attempt to improve i-vectors by better classifying the utterance information.

8.2.1 Bottleneck Features (BNFs)

Bottleneck features (BNFs) [Zhang et al., 2014] is one of the approaches to include DNNs in diarization. The idea behind bottleneck features is that MFCCs are not discriminative enough and should be substituted by a DNN crafted substitute. The obtention of the BNFs is done by means of a DNN senone recognizer, i.e. a DNN trained to identify senone acoustic units as part of an ASR task. This network must identify the senone ϑ_n, active at time n, according to the feature vector o_n and its surrounding context. The most popular architecture is a Multi Layer Perceptron (MLP), a monolithic construction of feed forward layers. Among these layers BNF extractors include a linear dimensionality reduction layer, also known as bottleneck, responsible to condensate the forwarded information. Given a trained DNN, the BNF extraction consists of the forward propagation from the DNN input up to the bottleneck, where the values at the neurons activations are considered as the new features, the BNFs. An example of this procedure is shown in Fig. 8.1, where the input o_n is classified in terms of the probability $P(\vartheta_n|o_n)$ and the BNFs are extracted from the dimensionality reduction.

Hybrid i-vectors based on BNFs are an alternative representation to the standard i-vector, only differing in the input feature, substituting MFCCs by BNFs. Nevertheless, this feature substitution can also be interpreted as an enhancement procedure by means of DNNs, making

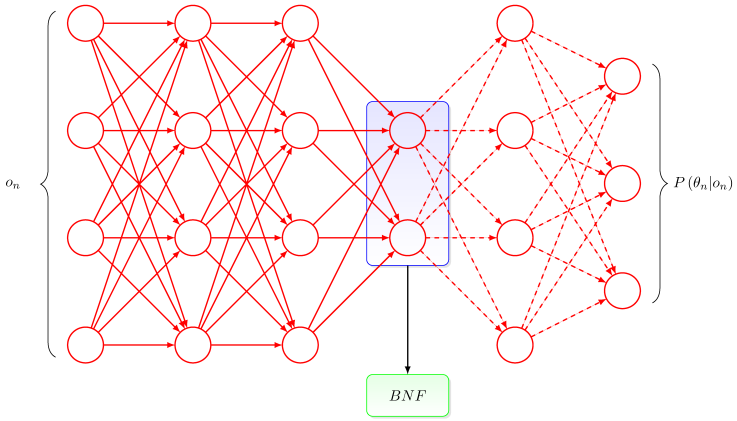


Figure 8.1: Example of a Bottleneck Feature extractor DNN

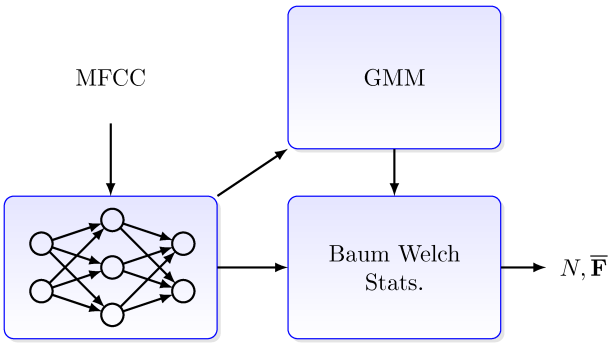


Figure 8.2: BNF pipeline from the original MFCCs up to Baum Welch statistics

the new crafted features more explanatory. Due to the fact that we are still working in terms of i-vectors, BNFs must follow the same pipeline MFCCs originally did, i.e. a trained GMM must infer the responsibility of the samples with respect to each component from the UBM-UBM. This GMM must be trained within the BNF domain, exclusive for each DNN setup. Once estimated these responsibilities we can calculate the zeroth and centered first Baum Welch statistics (N and $\overline{\mathbf{F}}$ respectively) also considering BNFs. Once N and $\overline{\mathbf{F}}$ are obtained, the extraction procedure is the same as with MFCCs. The procedure of extraction of both statistics from MFCC when BNFs are considered is shown in Fig. 8.2.

BNFs potential is caused by the senone classification by means of the DNN. During the classification process frames are sequentially transformed to highlight more and more the senone information, related to phonetics, and compensating undesired variabilities within the input data. Therefore, DNN transformations should make the different senones fit into more specific Gaussians, and making GMM components less variable.

8.2.2 Phonetic i-vectors

The phonetic i-vector concept [Viñals et al., 2019d] is an assignment strategy where DNNs collaborate with GMMs to assign features from an utterance along the UBM components. Given the utterance $O = \{o_1, ..., o_n, ..., o_N\}$ our goal is the improvement in the inference of the responsibility γ_{nc}, i.e. the estimation of the posterior probability of the sample n for being sampled from the cth component of the UBM.

In this proposal we seek improving the specificity of each component within the GMM-UBM. For this reason, we construct a GMM in which closed subsets of components are exclusively dedicated to the modeling of a context-independent acoustic unit. Then, when constructing the zeroth and first order Baum Welch statistics each frame only contributes to those components modeling the acoustic unit in the feature vector. Inferences about acoustic units in specific frames are done by means of a DNN, specifically trained to differentiate them.

An alternative interpretation of this approach works as a redefinition of the responsibility γ_{nc}. While the responsibility γ_{nc} in traditional i-vectors only requires the GMM-UBM, in the phonetic i-vector it is estimated by the composition of two different contributions. The first contributor is the GMM-UBM, which generates γUBM_{nc}. The second term is given by a DNN, defining γDNN_{nc}. This term is proportional to the classification probability $P_{DNN}(\vartheta_{kn}|O)$, describing how likely our DNN estimates the label ϑ_{kn} among K different phonemes according to the input sequence O. Mathematically:

$$\gamma_{nc} = \gamma UBM_{nc} \gamma DNN_{nc} \tag{8.1}$$

$$\gamma UBM_{nc} = \frac{\mathcal{N}(o_n|\boldsymbol{\mu}_c, \Sigma_c)}{\sum_{c'=1}^{C} \mathcal{N}(o_n|\boldsymbol{\mu}_{c'}, \Sigma_{c'})} \tag{8.2}$$

$$\gamma DNN_{nc} \propto P_{DNN}(\vartheta_{kn}|O) \tag{8.3}$$

One of the key points within the phonetic i-vector concept is the relationship between γDNN_{nc} and $P_{DNN}(\vartheta_{kn}|O)$. Alternative works working with responsibilities, such as [Lei et al., 2014], assume that the DNN differentiates among senons, associating each senon to a UBM component. Thus:

$$\gamma DNN_{nc} = P_{DNN}(\vartheta_{nc}|O) \tag{8.4}$$

By contrast, we relax this condition, only considering context-independent acoustic units. Hence, we reduce the number of classes to distinguish (K«C), while obtaining a more robust classification. Then, we define γDNN_{nc} as a mask where the DNN activates sets of components. Therefore, we define γDNN_{nc} as:

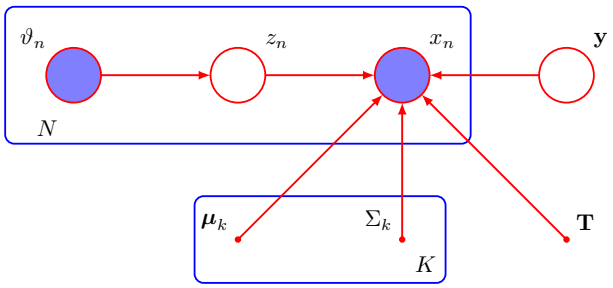


Figure 8.3: Bayesian network of the phonetic i-vector

$$\gamma DNN_{nc} = \begin{pmatrix} P_{DNN}\left(\vartheta_{1n}|O\right) \\ P_{DNN}\left(\vartheta_{2n}|O\right) \\ \vdots \\ P_{DNN}\left(\vartheta_{Kn}|O\right) \end{pmatrix} \otimes \underbrace{\begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}}_{C'} \tag{8.5}$$

where C' stands for an array of ones of length C' and $\otimes$ represents the Kronecker product. The result for this restriction is that the GMM has subsets of C' components to exclusively model each of the K phonemes recognized by the DNN. Moreover, we force the UBM to have KC' total components. By doing so, we get the power classification capabilities from DNNs and combine them with GMMs flexibility. This procedure can also be seen from the statistical perspective. From this point of view our UBM still follows a GMM distribution, but the assignment latent variable ϑ_n obeys to a prior information given by the DNN. The Bayesian network for the phonetic i-vector model is shown in Fig. 8.3.

Therefore, the i-vector extraction is constructed as follows. Given the sequence of features O, both the GMM and the phoneme classifiers receive the information as input. its result will be γUBM_{nc} and γDNN_{nc}, for all n samples in O. Afterwards, an extra block is responsible for its integration into γ_{nc}, constructing the zeroth and centered first order Baum-Welch statistics (N and $\overline{\mathbf{F}}$). These statistics will be used as input for the i-vector extraction as if a traditional i-vector was extracted. The pipeline procedure is illustrated in Fig. 8.4.

8.3 X-vectors

The last embedding proposal we study in this thesis is the x-vector, originally presented in [Snyder et al., 2016]. Based on a siamese neural network architecture, it was proposed as a speaker verification End-to-End (E2E) approach rather than an embedding extractor. However, after few iterations the most popular version for embedding extraction is defined in [Snyder et al., 2018]. The neural network is defined as a speaker recognition architecture in

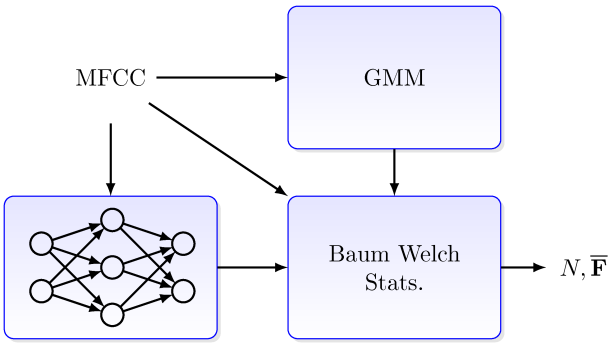


Figure 8.4: Phonetic i-vector pipeline from the original MFCCs up to Baum Welch statistics

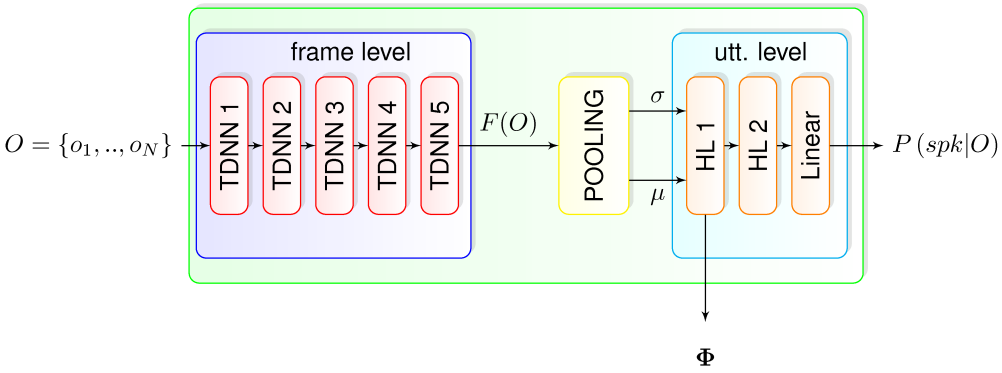


Figure 8.5: X-vector architecture schematic

which, given an utterance $O = \{o_1, ..., o_n, ..., o_N\}$, the network must decide among a closed set of candidate speakers. A graphical representation of the neural network is shown in Fig. 8.5.

The illustration in Fig. 8.5 divides the architecture in three main blocks. First, the input utterance is processed by a pool of Time-Delay Neural Network layers, a special type of convolutional layers only working in the temporal dimension and using dilations. Each convolutional network includes its linear transformation as well as the non-linearity and a Batch Normalization layer. Due to the fact that all these layers work in terms of frames and their context, we will refer to this block as frame-level block. The output for this level is a sequence of transformations $F(O) = \{f_1, ..., f_n, ..., f_{N'}\}$ where f_n are a transformation from the sample o_n and its contiguous context. In the original work this context includes approximately $150ms$ of speech. The second main block is the pooling block. The goal for this block is compaction of the utterance information along its temporal dimension, obtaining some utterance representation. In x-vectors architecture the representation is a stack of the mean $\boldsymbol{\mu}$ and standard deviation σ of the sequence $F(O)$. The final block takes the stack of $\boldsymbol{\mu}$ and σ as input, applying two fully connected hidden layers and a linear transformation just before classifying among the different

speakers according to the probability $P\left(spk|O\right)$. Again, each hidden layer includes its affine transformation as well as its non-linearity and a Batch Normalization layer. During training, the whole propagation is done up to the very last layer where cross entropy training cost is used. By contrast, during embedding extraction the forward propagation is extended up to the first layer within the utterance level block. There, the output for the affine transformation is our new embedding, also known as x-vector.

A more detailed analysis of the x-vector extraction procedure shows that, despite their conceptual differences, there are still some similarities between the new x-vectors and the standard i-vectors. These similarities involve both their extraction and posterior postprocessing. The first similarity is that both embeddings project the sequence O into a high-dimensionality space, obtaining the sequence $F(O)$ in the projected space. Besides, in both cases each one of the projections f_n takes into account the original o_n as well as its surrounding context. This context can either be generated by the succession of convolutional layers (x-vector) or the application of derivatives (i-vectors). Secondly, we must talk about the pooling layer. I-vectors model the latent variable explaining the utterance O in terms of a Gaussian distribution, parametrized in terms of its mean $\boldsymbol{\mu}$ and covariance Σ. Similarly, x-vectors essentially model the projected sequence $F(O)$ as an undetermined distribution, but characterized by the same first and second central moments. A final clear similarity between x-vectors and i-vectors postprocessing can also be observed. In previous chapters we described the Uncertainty Propagation concept (Chapter 5), where the i-vector is compensated by the Cholesky decomposition of its covariance. In x-vectors we are applying a similar concept during the first layer at the utterance level block. There, the estimated mean $\boldsymbol{\mu}$ is linearly transformed and an extra term is added, based on the linear transformation of the standard deviation, an approximation of i-vectors $\Sigma^{1/2}$.

8.4 Experiments

After the theoretical explanation of the different DNN embeddings, the purpose of the following lines is the study of their performance in a diarization task. However, due to software issues not all the embeddings could be tested on the same conditions. For this reason, we first present those results obtained with bottleneck features (BNFs), later showing the performance of both phonetic i-vectors and x-vectors, which share a common experimentation scenario.

8.4.1 Bottleneck Features (BNFs)

The study of bottleneck features in broadcast diarization follows [Viñals et al., 2016], which analyzes their inclusion in diarization making use of the MGB 2015 dataset according to the description in Section 3.2.1. These experiments consider an alternative diarization architecture, described in [Villalba et al., 2015]. This architecture takes into account the baseline system described in Section 4.1.2, although it performs a triple step FBPLDA resegmentation, with an i-vector reevaluation after each iteration. Nevertheless, the system setup, a 256 component GMM-UBM with a 100-dimension T matrix and a 50-dimension FBPLDA model, remains unmodified compared to the thesis baseline. This system will be trained and evaluated with two types of features, MFCCs as baseline and BNFs as alternative.

For the BNF extraction, we take into account a Multi-Layer Perceptron network, inferring a senone label ϑ_n in terms of a pool of 23 frames ($\{o_{n-11}, ..., o_{n+11}\}$). Our analysis studies architectures with different size (5-7 layers and 256-2048 neurons per layer) as well as the bottleneck location (only after a non-linear layer up to just before senone classification) and size (50-100). The network is trained using senone alignment labels estimated by an HMM-GMM strategy by means of the Kaldi toolkit [Povey et al., 2011]. The number of involved senones is around 4000. Because the tuning for this HMM-GMM falls beyond the scope of this thesis, a baseline setup released by MGB organization was considered instead [Bell et al., 2015].

For the study of the best BNF architecture we did not study the whole totality of the hyper-parameter space. Instead, we relied on Bayesian optimization techniques [Snoek et al., 2012], exploring those setups more likely to improve the performance. Along the experimentation stage, the results have revealed few informative patterns of interest. The first interesting result is about the location of the bottleneck layer, i.e. its position along the sequence of feed forward layers. In order to provide a more rigorous analysis we will consider the pool of results we obtained during our experiments, averaging those results with respect to the bottleneck location (i.e. number of layers, their size and the bottleneck size). This location in our experiments can be placed after 1, 2, 3 or all non-linear layers. The obtained results with MGB 2015 evaluation set can be seen in Table 8.1.

Table 8.1 highlights the relevance for the Bottleneck location. According to the obtained results, the extraction procedure should be done as soon as possible during the forward propagation. This result seems reasonable despite our previous analysis. It is true that DNNs can delete undesired variabilities in posterior layers, theoretically benefiting our goal. Nevertheless, we must also bear in mind that the phonetic variabilities we use to discriminate speakers are harmful for this DNN. Thus, the sooner is the bottleneck the less likely the DNN can delete this

Number of previous non-linear layers	Average DER(%)	
	Dev. Subset	Eval. Subset
1	26.71	**47.04**
2	**24.76**	49.98
3	28.64	49.96
Last	36.57	56.16

Table 8.1: DER (%) results in terms of the bottleneck layer position along the DNN. Tested after 1, 2,3 non-linear layers as well as right before the final classification linear layer. Results obtained with MGB 2015 evaluation subset

Type of feature	Average DER(%)	
	Dev. Subset	Eval. Subset
FB	**26.17**	**45.15**
MFCC	26.91	47.04

Table 8.2: DER (%) results in terms of the features for BNF extraction. MFCCs and Filter Bank features considered. Results obtained with MGB 2015 evaluation subset

valuable information.

Another idea we tackled was the feature input to the DNN. While MFCCs were the most popular approach, we could also test alternative features. For this purpose, we compare those bottlenecks obtained from MFCCs with those extracted from Filter Bank (FB) features. These last features follow the same extraction procedure MFCCs do, except for the fact that no final DCT is carried out. For this new comparison we analyze those results involving both input features but restricted to have the bottleneck after the first non-linear layer. Again, we average the results obtained for each type of feature. Table 8.2 shows the results for the comparison, carried out on MGB 2015 evaluation subset.

The average results in Table 8.2 indicate that Filter Bank features provide a significant benefit in terms of performance when compared to MFCCs. Due to the differences among both types of features, it looks like DNNs struggle to deal with correlated data in the first stages of the forward propagation. This difficulty may preserve the useful i-vector information at the bottleneck location and thus improving the performance.

After having explored two of the most noticeable obtained results during our experimental search, we now want to present the comparison between BNFs and MFCCs. In this comparison we have chosen those best DNN setups for both MFCC and FB inputs. The comparison is illustrated in Table 8.3.

Experiment	DER(%)	
	Dev. Subset	Eval. Subset
MFCC	28.35	42.45
BNF (FB)	**25.13**	45.22
BNF (MFCC)	25.90	**45.07**

Table 8.3: DER (%) results for MFCCs and BNFs evaluating MGB 2015 evaluation subset

Experiment	DER(%)	
	Dev. Subset	Eval. Subset
MFCC	28.35	42.45
BNF (FB) + MFCC	**23.61**	**42.25**

Table 8.4: DER (%) results for BNF and MFCC fusion.

Unfortunately, the comparison of performance shown in Table 8.3 reveals that no bottleneck feature is able to outperform MFCCs for the evaluation subset, despite development results evidence the opposite. This behaviour is shared by the two types of bottlenecks, those obtained by means of MFCCs and those obtained from FB.

For this reason, we analyze the option of fusing both types of embeddings, standard MFCCs and BNFs, as shown in [Lozano-diez et al., 2016][Hamidi Ghalehjegh and Rose, 2015] [Viñals et al., 2016]. Then, for the same experimental setup the obtained results are those shown in Table 8.4

The results in Table 8.4 show that the joint work of both features manages to overcome simple MFCCs. However, the benefit is almost negligible. Interestingly, the results for the development set with our new features significantly improve those obtained with MFCCs, revealing some of their potential.

8.4.2 Phonetic i-vectors & x-vectors

In addition to i-vectors constructed with bottleneck features, we can also use alternative embeddings as phonetic i-vectors and x-vectors. Due to the complexity of their inclusion in diarization, we opted for gaining insight about this technology prior to its application on the target task. For this purpose we analyze both embeddings in a speaker recognition task, applying the acquired knowledge in diarization.

8.4.2.1 Speaker recognition

The speaker recognition scenario we opt to analyze the new embeddings follows those already described in Section 5.2.1: The SRE10 coreext-coreext det 5 female experiment is analyzed by an embedding-PLDA system trained on the allowed corpus (SRE04, 05, 06 and 08). Along the following experiments we will compare the performance of traditional i-vectors, obtained by means of a GMM-UBM, with the new embeddings. Moreover, the experiments will cover an scenario with long utterances, using the original utterance set, as well as an alternative scenario that considers short utterances. This latter experiment is expected to be closer to diarization conditions.

The baseline system, also described in Section 5.2.1, consists of a 2048-component GMM-UBM followed by a 400-dimension T matrix. The extracted i-vectors are dimensionally reduced to 200, feeding a 200-dimension PLDA afterwards. Scores undergo neither score normalization nor calibration.

The first embedding to evaluate is the phonetic i-vector. Our alternative approach takes into account a 2496-Gaussian GMM (64 components per phoneme, with 39 phonemes), using as input both the features as well as the acoustic unit labels. This GMM jointly works with an acoustic unit DNN classifier responsible for providing the input labels. This network, defined in [Viñals et al., 2019d], is based on Wide Residual Networks [Zagoruyko and Komodakis, 2016], also known as WideResNet or WRN. The network structure can be interpreted as 4 blocks of layers. Each block consists of a TDNN imitating the Shifted Delta Cepstral (SDC) [Calvo et al., 2007] and 4 WRN layers, including 2 convolutional layers per WRN construction. Each involved convolutional layer is composed by the linear computation as well as the nonlinearity (ReLU) and a Batch Normalization layer [Ioffe and Szegedy, 2015]. A last linear layer is responsible for weighing the output from the last block into each one of the target phonemes. While the GMM is exclusively trained on excerpts from SRE04, 05, 06 and 08, the DNN considers Fisher and Switchboard datasets. The labels for the DNN training are obtained by means of an HMM-GMM procedure using the Kaldi toolkit [Povey et al., 2011]. Regarding the remaining elements in the recognition system, all they share the baseline training configuration.

Another alternative to i-vectors and phonetic i-vectors are x-vectors. In order to test their performance we opted for the original setup described in [Snyder et al., 2016], but exclusively trained on the same corpus the baseline does, i.e. SRE04, 05, 06 and 08. No data augmentation is considered in the training process. Apart from the embedding extraction technique, x-vectors require a different backend configuration. Thus, the resulting x-vectors, with 512 dimensions

Experiment	EER (%)	minDCF
Long-Long Scenario		
Original I-vector	3.37	0.16
PHN I-vector	**2.33**	**0.12**
X-vector	2.40	0.15
Short-Short Scenario		
Original I-vector	8.76	0.40
PHN I-vector	7.23	0.37
X-vector	**6.93**	**0.35**

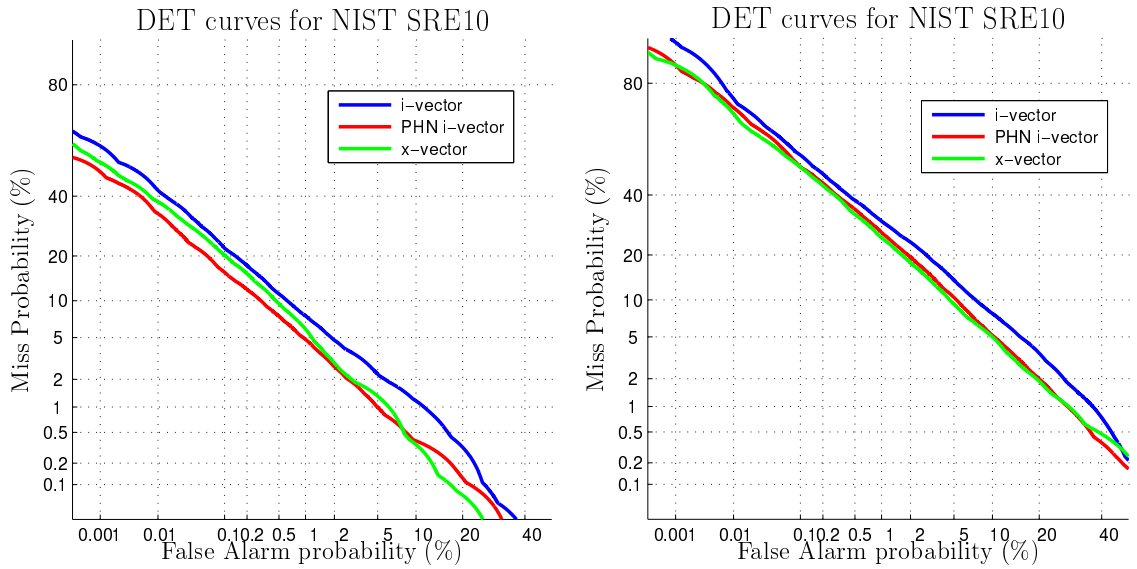
Table 8.5: Phonetic i-vector and x-vector performances in speaker verification with SRE10 coreext-coreext det5 female. Measured both EER (%) and minDCF. Experiment carried out with the original long utterances and the chopped short version

according to their original publication, will be later compressed by means of LDA into a 200 dimension subspace. Then, the SPLDA backend will compress even more the speaker information within a 50-dimension subspace.

The performance of both new embeddings compared to the traditional i-vector is illustrated in Table 8.5. Two different performance metrics, EER and minDCF are evaluated. This experiment includes both the evaluation of the long original utterance subset as well as their shortened version.

According to Table 8.5, the substitution by the standard i-vector by means of the phonetic i-vector implies a small but consistent improvement of performance, affecting the two metrics EER and minDCF. X-vectors also show a significant improvement with respect to the traditional i-vectors. This improvement is consistent, affecting both metrics. Both embeddings, phonetic i-vectors and x-vectors, show a better performance regardless of the utterance length. Interestingly, phonetic i-vectors behave better than x-vectors when long utterances are involved, while short utterances are better treated by x-vectors. It is mandatory to clarify that the x-vector training data pool was the same for both systems, despite limited for DNN training purposes. In general x-vectors include in its training much more information, e.g. [Viñals et al., 2019d], where x-vectors outperform phonetic i-vectors in telephone channel. However, in spite of this data limitation x-vectors succeed in outperforming their phonetic i-vectors counterparts when evaluating short utterances. The obtained results can be complemented by the whole DET curve, shown in Fig. 8.6, where we represent traditional i-vectors (blue), phonetic i-vectors (red) and x-vectors (green).

The DET curves in Fig. 8.6 are consistent with those results shown in Table 8.5. Pho-



a) Long utterances b) Short utterances

Figure 8.6: DET curves for x-vectors in SRE10 with long and short utterances

netic i-vectors evidence a consistent improvement with respect to the standard i-vectors along the curve. This improvement affects both studied scenarios, Long-Long and Short-Short trials. Nevertheless, according to the curves, benefits are less relevant when short utterances are involved. Regarding x-vectors, they also outperform the traditional i-vector regardless of the operating point or the utterance length. Moreover, the low amount of training data for x-vectors makes phonetic i-vectors outperform them for long utterances along the whole curve. By contrast, when short utterances are evaluated, differences are minimized, slightly outperforming the x-vector approach.

8.4.2.2 Broadcast diarization

Thanks to the previous experiments we have observed that the introduction of DNNs into the characterization of speakers in the telephone domain have improved those systems exclusively relying on purely statistical methods, e.g. the i-vectors. Besides, these improvements are noticeable in short utterance conditions, typical in diarization conditions. Nevertheless, when moving towards a more variable domain, such as broadcast, and a different task, diarization, the performance might significantly differ. In this section we analyze the performance of phonetic i-vectors and x-vectors in the diarization task.

For these experiments we consider the evaluation of Albayzín 2018, according to the description in Section 3.2.2. Besides, the baseline diarization system follows the description in

Section 4.1.2, considering a 256 Gaussian GMM-UBM, followed by a 100-dimension T matrix. The estimated i-vectors are whitened by means of PCA prior to length normalization. The final backend is a 100-dimension FBPLDA.

The first embedding to evaluate is the phonetic i-vector. The extraction for this embedding slightly differs from the configuration presented for the speaker recognition task. For diarization we build a phonetic GMM that maps 34 phonemes, each one modeled by 8 exclusive Gaussians (272 Gaussians in total). The acoustic labels, required for the phonetic i-vector, are provided by a DNN classifier, whose architecture is similar to the previously seen in the previous speaker verification experiment. Differences are reduced to consider 5 blocks of two WRN layers, each one including 2 convolutional networks consisting of a linear computation, the non-linearity and a Batch Normalization layer. The whole network is trained on RSR2015 [Larcher et al., 2012], Timit [Garofolo et al., 1993], Librispeech [Panayotov et al., 2015], Tedlium [Rousseau et al., 2012] and Voxforge, using as target phonemes labels those obtained by an HMM-GMM from the Kaldi toolkit. The remaining elements of the system are exclusively trained with Albayzín 2018 available corpora. The use of data outside Albayzín evaluations makes this solution to be part of the open-set condition of Albayzín 2018.

Moving to x-vectors, their great performance in speaker verification, even in short utterances where they still improve the phonetic i-vector alternative, motivates its inclusion in diarization. For their evaluation in broadcast diarization we use of a trained online available network [1] working on the Kaldi toolkit. This network is trained using VoxCeleb I [Nagrani et al., 2017] and II [Chung et al., 2018], considering data augmentation too. This network follows the originally published setup, extracting 512-dimension x-vectors. The remaining diarization system, exclusively trained with Albayzín 2018 corpus, according to the description in Section 3.2.2, follows a similar setup as the one used in speaker verification. The extracted embeddings are dimensionally reduced by means of LDA (200 dimensions), undergoing a final subspace compression by means of PLDA (50 dimension). Due to the use of external data for the training of the x-vector extractor, this system also lies as open-set for the original evaluation purposes.

The results for the comparison of the new embeddings and the standard i-vectors are presented in Table 8.6. The experiment involves two different backends, the FBPLDA VB resegmentation and the PLDA sequential tree approach.

The results in Table 8.6 show consistent improvements (around 10% relative improvements) in the diarization task when phonetic i-vectors are taken into account. This reduction of the DER term is obtained along both types of clustering approaches, indicating the consistency of

[1] http://kaldi-asr.org/models/m8

Embedding	DER(%)
FBPLDA VB resegmentation	
i-vector	17.79
phn i-vector	**16.05**
x-vector	**15.93**
Sequential trees	
i-vector	17.60
phn i-vector	**15.39**
x-vector	**13.89**

Table 8.6: DER(%) results for the original i-vectors, phonetic i-vectors and x-vectors with Albayzín 2018. Clustering performed with both FBPLDA and sequential tree-based clustering.

the results with the new embedding. Better results are obtained (at least 10% relative improvements) whenever x-vectors are considered. These benefits are obtained with both clustering approaches, although sequential trees reveal a much more significant improvement (up to 21% relative improvements). Thus, x-vectors outperform phonetic i-vectors with both clustering approaches. Besides, this extra improvement is more noticeable with the sequential tree, being for the FBPLDA almost negligible.

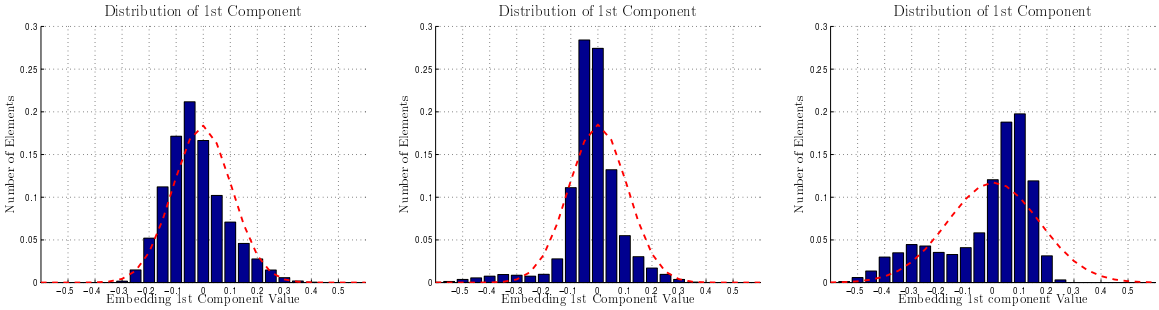
According to the obtained results in both telephone channel and broadcast diarization, the inclusion of DNN-based embeddings clearly improve the performance. However, while speaker recognition strategies in telephone channel show x-vector outperforming hybrid i-vectors, in our previous examples this situation did not happen. In fact, depending on the considered clustering technique, the performance for both embeddings was very similar. Thus, we should understand the reasons why certain embeddings behave better.

For this purpose, we study the distribution of the embeddings right before feeding the PLDA model. at this point, embeddings have already undergone the centering, whitening and length normalization processing. According to its definition, PLDA requires embeddings to be standard normal. In order to fulfill this condition, the distribution of each component of the embedding must follow a Gaussian distribution as well. For this purpose, we evaluate a Gaussianity test on the different dimensions of the embeddings, the Kolmogorov-Smirnov test, evaluating whether each dimension of the embeddings fit a Gaussian distribution. In Table 8.7 we summarize the counting, illustrating the proportion of dimensions Normally distributed according to the Kolmogorov-Smirnov test, at a significance level of 5%.

According to Table 8.7, none of the evaluated embeddings are close to be considered as a

Embedding	Percentage of Gaussian distributed dimensions (%)
I-vector	30
Phonetic i-vector	34.50
X-vector	19.50

Table 8.7: Proportion of Gaussian dimensions on embeddings according to Kolmogorov-Smirnov test. The null hypothesis is not rejected at the 5% significance level.



a) I-vector b) Phonetic I-vector c) X-vector

Figure 8.7: Distribution of the embedding first component in standard i-vectors, phonetic i-vectors and x-vectors for the training corpus in Albayzín 2018. Represented the histogram and the Gaussian distribution fitting the data in red.

pure Gaussian distribution. Nevertheless, both i-vector embedding types present at least 50% extra Gaussian dimensions compared to x-vectors, making them closer to satisfy the request. In addition to numbers, visual analysis also illustrates how x-vectors are less Gaussian. Taking a look to specific dimensions of the embeddings we can represent their distributions as those shown in In Fig. 8.7, where we illustrate the distribution of the first component of standard i-vectors, phonetic i-vectors and x-vectors for Albayzín 2018 training subset.

The joint consideration of Table 8.7 and Fig. 8.7 indicates the non-Gaussian nature of the extracted embeddings. A consequence for this observation is that our clustering systems, all of them relying on this Gaussian assumption, may suffer from degradations, the more severe as long as the embedding is further from a Gaussian distribution (x-vectors).

8.5 Conclusions

The experimental work developed along this chapter has evidenced the benefits due to the inclusion of DNNs in the embedding extraction. These benefits have been observed along the three types of new embeddings, although only the phonetic i-vectors and x-vectors could significantly

improve in the evaluation set.

Regarding hybrid i-vectors based on bottleneck features, our results indicate an important mismatch between development and test. Thus, improvements obtained in development (around 10% relative improvement) could not be observed in the evaluation subset. Moreover, our experiments highlight the relevance of the bottleneck position along the network, being convenient its location after the first hidden layer. In addition to that, our experiments also reveal that Filter Bank features are able to outperform the more complex counterparts. Finally, MFCCs and Bottleneck features are complementary types of information, being able to jointly collaborate for an improvement in performance.

With respect to phonetic i-vectors and x-vectors, our experiments in speaker recognition evidence the benefits of DNNs in embeddings, always outperforming i-vectors regardless of the operating point. Moreover, this benefit is consistent when tested for short utterances, a closer scenario to diarization. In fact, in this situation x-vectors slightly outperform phonetic i-vectors despite not exploiting all the modeling capabilities. When tested in diarization both embeddings are able to clearly outperform traditional i-vectors with two clustering approaches developed along the thesis. Nevertheless, the best benefits are always obtained with x-vectors, specially when evaluating the PLDA-based sequential tree approach.

According to our understanding, this different behaviour of performance between phonetic i-vectors and x-vectors in diarization depending on the clustering approach is very related to the Gaussian hypothesis our models assume. Despite the fact that no tested embedding can be considered Gaussian, x-vectors present the furthest distribution to a standard normal. Thus, clustering solutions highly relying on distributions, as the VB FBPLDA reclustering, might suffer from stronger degradations compared with other alternatives such as the Tree Sequence approach.

Part IV

The Model Adaptation Problem

Chapter 9

Data-Efficient Domain Adaptation for PLDA Models

Along the thesis we have seen how different techniques have managed to improve the overall performance of a diarization system. By focusing on different subtasks, such as the clustering block (Chapters 4, 5 and 6) and the embedding extraction (Chapters 7 and 8) we managed to obtain significant average improvements. Nevertheless, those results in Table 3.3 or Table 3.5 show a very different behaviour of the diarization system depending on the evaluated show. Considering that each show or genre presents individual characteristics, we may consider them as different domains, usually unseen during training.

For this purpose, this chapter is dedicated to the adaptation of diarization systems to unseen domains specially when available resources are limited or simply unavailable. This situation is very common in broadcast data where it is almost impossible to cover all shows or genres.

9.1 Introduction

Diarization nowadays has become a very popular task, with multiple scenarios requiring its application. Unfortunately, performance is very dependent on the evaluation scenario, obtaining good performance if the domain during evaluation matches the training one, severely degrading otherwise. A suitable solution is the training of multiple systems, each one specially constructed for each domain. However, this option requires the training of expensive systems in terms of time and data. Besides, this option is very rigid, suffering from new unseen scenarios or domains where not enough data is available. Recently, speaker verification domain independent approaches [Rohdin et al., 2019] [Nidadavolu et al., 2019] have been proposed. These systems are trained to project the information within the same speaker spaces regardless their original do-

main. Nevertheless, this option also lacks of flexibility when unknown scenarios are presented to the system.

Therefore, an intermediate solution is model adaptation. The key point is the generation of a single model or pipeline of systems, all trained with the very best state-of-the-art techniques and no restrictions regarding time and data. Once this pipeline is trained, only few of its components can be adapted to deal with any given scenario. For this purpose, systems only need few in-domain data, much less than a whole training corpus.

To reduce domain mismatch, modern diarization systems require in-domain data to adapt their models. Nevertheless, when these in-domain data are scarce, domain mismatch can only be handled by unsupervised adaptation techniques. This concept is analyzed in [Le Lan et al., 2016] [Viñals et al., 2017] [Viñals et al., 2019c], where models are successfully adapted using unlabeled in-domain data. This option let us replace expensive hand-transcribed data by automatically obtained pseudo-speaker labels.

9.2 Methods for domain mismatch reduction

PLDA performance is known to suffer from strong degradation when facing a domain mismatch between training and evaluation conditions. The same kind of mismatch we first observed in Section 3.3 for Broadcast data when studying the differences among episodes, shows and genres. The large number of different domains makes training particular models unfeasible, so domain adaptation is the best option.

Adaptation in models with speaker awareness (e.g. PLDA) requires some speaker labels θ_{ADAPT}, as illustrated in Fig. 9.1a. This is also referred as supervised adaptation. However, in many situations perfect labeled in-domain data is either limited or just unavailable. For those situations, in [Viñals et al., 2017] it was proposed the unsupervised adaptation with pseudo-speaker labels (Fig. 9.1b). The necessary speaker labels θ_{ADAPT} were estimated only considering the evaluation data.

This strategy can be interpreted as a dual diarization. First, it is performed a diarization step on the data themselves by means of naive techniques, with a limited or null knowledge of speaker variability. The first step infers the pseudo-speaker labels considered for model adaptation, adapting the global PLDA model to the specific domain. The new specific model is then used to perform the final diarization.

In [Viñals et al., 2017] it was only presented the basic unsupervised adaptation block, which estimated the pseudo-speaker labels by means of naive clustering techniques. These pseudo-speaker labels, known not to be totally reliable, were refined during the PLDA adaptation, con-

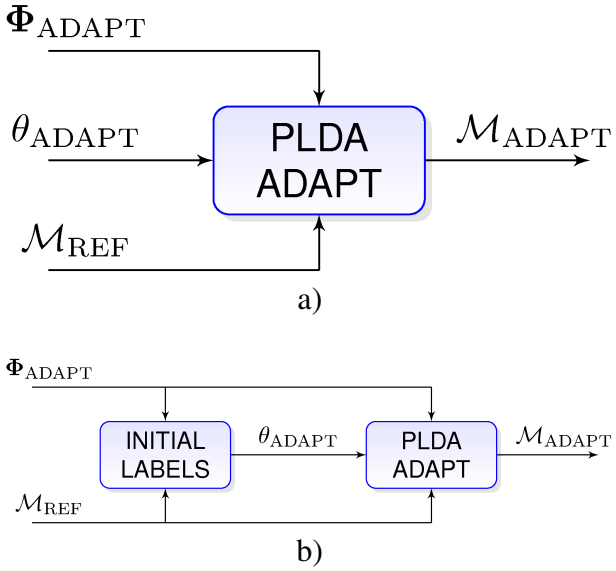


Figure 9.1: Schematic for the a) supervised and b) unsupervised adaptation

sidering them another trainable parameter.

We propose multiple approaches based on this unsupervised adaptation block to deal with domain mismatch, analyzing the robustness of the process. For this purpose, we first study the impact of speaker awareness in the pseudo-speaker label estimation. Two main options, cosine similarity, which lacks of any knowledge about the speaker subspace, and PLDA likelihood ratio are tested. Efficient clustering techniques are considered, such as Agglomerative Hierarchical Clustering (AHC) and Mean Shift [Fukunaga and Hostetler, 1975] [Senoussaoui et al., 2014] [Salmun et al., 2017] [Stafylakis et al., 2010]. The estimated pseudo-speaker labels are evaluated in multiple adaptation approaches, including totally unsupervised strategies when perfectly labeled in-domain data is unavailable and semi-supervised alternatives when these data are just scarce. All these approaches are validated by direct comparison with the traditional supervised adaptation, performed with the same limited data. The proposed modalities are very oriented to broadcast scenarios, where the tradeoff between expenses and labeled resources is an important factor in decision making. The proposed alternatives are:

- *Independent unsupervised strategy*

  Our first proposal is the independent unsupervised adaptation strategy, which individually performs the adaptation, episode by episode. A conceptual representation is illustrated in Fig. 9.2. For each episode n we adapt the out-of-domain PLDA model $\mathcal{M}_{\text{OOD}}$ only taking into account the i-vectors $\mathbf{\Phi}_n$ from episode n. The result is the adapted model $\mathcal{M}_n$.

- *Longitudinal unsupervised strategy*

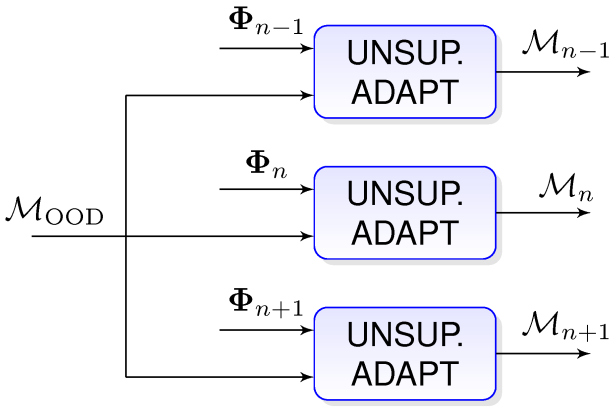


Figure 9.2: Schematic for unsupervised independent adaptation for the episodes $n-1$, n and $n+1$.

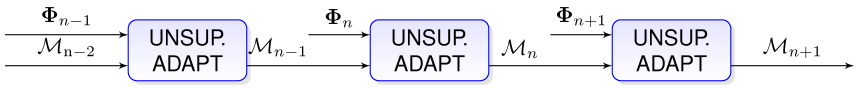


Figure 9.3: Schematic for unsupervised longitudinal adaptation for the episodes $n-1$, n and $n+1$.

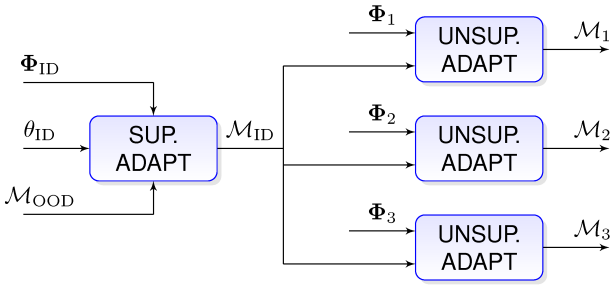


Figure 9.4: Semi-supervised adaptation strategy based on the unsupervised independent adaptation approach for the episodes $n-1$, n and $n+1$.

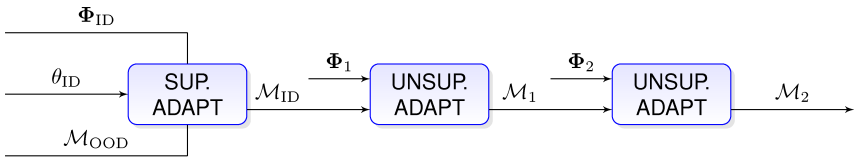


Figure 9.5: Semi-supervised adaptation strategy based on the longitudinal unsupervised adaptation approach for the episodes $n-1$, n and $n+1$.

Broadcast content from a show usually involves multiple episodes (i.e. a season). These multiple episodes are *a priori* likely to have similar acoustic information (same speakers and similar acoustic conditions). Therefore, we can take into account more than one episode to perform the PLDA adaptation, supervised or not. In the longitudinal approach episode n is adapted considering the result of the adaptation $\mathcal{M}_{n-1}$ from the previous episode $n-1$ as reference model. This strategy is illustrated in Fig. 9.3. By this way, we expect that successive adaptations could retain show-dependent information to improve the performance.

- *Independent semi-supervised strategy*

  We also propose semi-supervised architectures, assuming that few labeled data are available. In real applications a perfectly labeled small subset of data may be available. Therefore, we want to test whether we can combine the knowledge acquired from a small subset of supervised data (e.g. one or two episodes) with the one obtained by the unsupervised adaptation.

  Our first hybrid proposal considers a model adaptation stage in terms of the supervised labeled data followed by an unsupervised domain adaptation, independent for each episode. In this approach the out-of-domain model $\mathcal{M}_{\text{OOD}}$ is first adapted with the in-domain perfectly labeled data, obtaining the in-domain model $\mathcal{M}_{\text{ID}}$. This model is later specifically adapted to each episode using the unsupervised adaptation block. The architecture is shown in Fig. 9.4.

- *Longitudinal semi-supervised strategy*

  We also test a semi-supervised longitudinal strategy when dealing with multiple episodes from the same show. The out-of-domain model $\mathcal{M}_{\text{OOD}}$ is supervisedly adapted with the limited labeled data (i-vectors $\mathbf{\Phi}_{\text{ID}}$ and labels θ_{ID}), generating an in-domain model $\mathcal{M}_{\text{ID}}$. This model is then unsupervisedly adapted in a longitudinal way, i.e., the resulting adapted model for episode n will work as reference model for episode $n+1$. Its schematic is illustrated in Fig. 9.5.

9.3 Experiments

Once the different approaches to evaluate are explained, this section proceeds to its evaluation. For this purpose, we will make use of the diarization system presented in Section 4.1.2. This

system will play the role of baseline system without any sort of adaptation. The experiments will be carried out with MGB 2015 according to the description in Section 3.2.1.

9.3.1 Independent unsupervised adaptation

Our first experiment compares those results obtained with the baseline system and shown in Chapter 4 with those obtained when we carry out the novel independent totally unsupervised adaptation strategy. We propose exploring the four possible pseudo-speaker labels initializations described in section [Viñals et al., 2017]: Two clustering modalities, Agglomerative Hierarchical Clustering (AHC) and Mean-shift (MS) working with two similarity metrics, cosine similarity (COS) and PLDA log likelihood ratio (PLDA). The results for this experiment are shown in Table 9.1.

Table 9.1: DER(%) for the unsupervised adaptation in the evaluation set.

ADAPT LABELS	AHCPLDA	AHCPLDA+VBPLDA
No adaptation	49.39	41.58
AHC COS	41.16	39.01
MS COS	**40.08**	**34.95**
AHC PLDA	44.39	44.36
MS PLDA	43.15	41.79

The comparison of results from Table 9.1 show the benefits of the unsupervised adaptation. The first step in the diarization system (the Agglomerative clustering with PLDA llr) evidences approximate 10-20% relative improvements when adapted models are considered, regardless the pseudo-speaker labels. Besides, these results are improved by means of the Variational Bayes refinement (VBPLDA resegmentation), also considering the adapted models. However not all the pseudo-speaker labels are equally useful. Some of these labels lead to local DER minima from which the Variational Bayes posterior resegmentation does not provide any extra improvement.

All the experiments with cosine similarity pseudo-speaker labels have outperformed the PLDA-based counterparts and perform better than the baseline. In fact, PLDA based pseudo-speaker labels are harmful for adaptation purposes, getting degraded with respect to the baseline results. Moreover, in all cases Mean-Shift has obtained better results than the Agglomerative Hierarchical Clustering.

Table 9.2: DER(%) results for the unsupervised adaptation with longitudinal model propagation in the evaluation set.

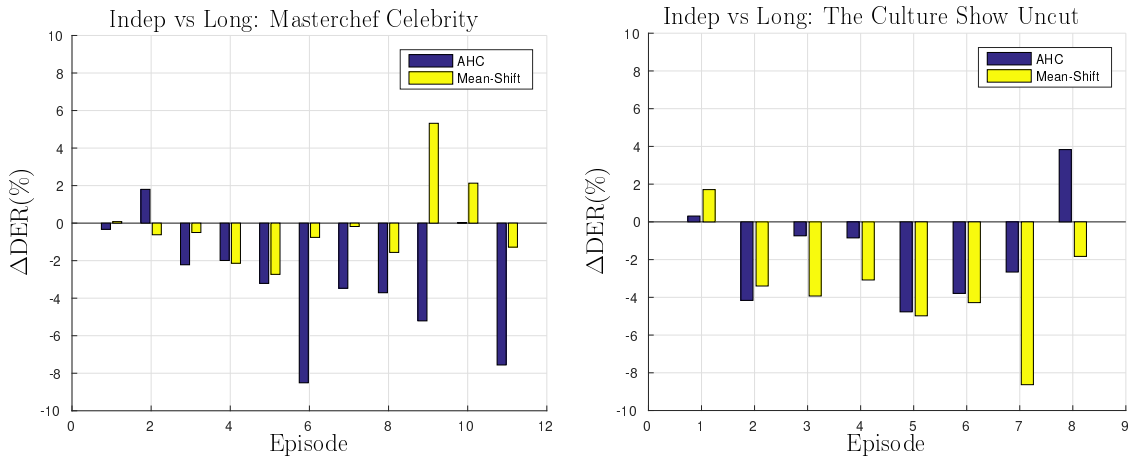
EXPERIMENT	DER(%)
BASELINE	41.58
AHC COS	41.46
MS COS	**36.27**

9.3.2 Longitudinal unsupervised adaptation

Some of the results included in Table 9.1 show a significant improvement respect to our baseline. This improvement is obtained despite considering a small amount of in-domain information (up to one hour of audio). Taking into account multiple episodes (all the episodes from a show) with our longitudinal proposal we expect to get bigger improvements. Table 9.2 shows the results of the longitudinal unsupervised adaptation approach for the evaluation set. Agglomerative Clustering (AHC) and Mean-shift (MS) are studied with cosine similarity (COS). The longitudinal adaptation is done along all the episodes from a show.

The results in Table 9.2 also outperform the reference, especially when Mean-Shift is used. However, as in the independent adaptation, the agglomerative clustering behaves significantly worse than Mean-Shift. For both cases, agglomerative clustering and Mean-Shift, the longitudinal unsupervised adaptation shows a small degradation versus the independent counterpart. This small degradation can be attributed to the consecutive adaptations with noisy data. In consequence it is important to determine if this longitudinal strategy overcomes the independent one considering less episodes in a row. For this reason, we analyze the results episode by episode, shown in Fig. 9.6. We illustrate the difference between DER results obtained with the independent approach versus the longitudinal one ($\Delta_{\mathrm{DER}} = \mathrm{DER}_{\mathrm{INDEP}} - \mathrm{DER}_{\mathrm{LONG}}$) for each episodes from both shows.

Fig. 9.6 reveals that the longitudinal approach compared to the independent adaptation suffers from a degradation which affects similarly all the episodes. Besides, the analysis indicates this behavior is shared for both the Agglomerative hierarchical pseudo-speaker labels and the Mean-Shift ones. The results indicate that the degradation already appears in the second episode from both shows. Therefore, a longitudinal adaptation in few episodes is not expected to take any advantage.



a) Show 6: Masterchef Celebrity

b) Show 7: The culture show uncut

Figure 9.6: ΔDER (%) performance episode by episode for the two shows of the evaluation set. Defined as $\Delta\text{DER} = (\text{DER}_{\text{INDEP}} - \text{DER}_{\text{LONG}})$. AHC refers to the Agglomerative clustering pseudo-speaker labels.

9.3.3 Use of in-domain labeled data and semi-supervised adaptation

Our previous experiments have reported a significant improvement of the DER measure due to the unsupervised adaptation with pseudo-speaker labels, especially with those created with Mean-Shift and cosine similarity. However, we cannot compare these results with the traditional supervised adaptation because MGB dataset does not provide extra in-domain labeled data for this purpose. Therefore, we propose an alternative dataset arrangement. We divide the evaluation set into two parts. The first one is dedicated to supervised adaptation, containing the first episode from each show to evaluate. The new evaluation subset contains all the remaining episodes from the same shows. This modification of the evaluation subset makes unfair any comparison with the previous results and those obtained in the original MGB 2015 challenge. Hence both the baseline system as well as the fully unsupervised approaches must be reevaluated.

With the new distribution of data, we compare the classical supervised adaptation, with our new proposed alternatives, both the independent and the longitudinal approach. In this experiment we have evaluated supervised adaptation with only one-hour episode for each show as in-domain information. The results are shown in Table 9.3.

The results in Table 9.3 show that our proposed unsupervised adaptations (independent and longitudinal approaches) outperform the supervised adaptation with the baseline system when few in-domain data are used (1-hour episode from each show). Again, the independent unsupervised adaptation approach gets the best results, obtaining up to 9% relative improvement.

Table 9.3: DER (%) results of supervised and unsupervised (independent and longitudinal) adaptation with the new data distribution in the evaluation set.

Adaptation	DER(%)
Baseline	41.65
Supervised	39.00
Unsup. Independent	**35.39**
Unsup. Longitudinal	37.00

This result is specially noticeable because in-domain information automatically estimated from the data we are diarizing can be more informative than small amounts of manually annotated in-domain data.

The new data distribution provides perfectly labeled in-domain audio. Hence semi-supervised approaches can also be analyzed, first applying some supervised adaptation of the models with the available labeled data and then unsupervisedly adapt to the evaluation audio. In Table 9.4 we compare the baseline system with respect to all the proposed adaptation techniques (supervised, unsupervised independent, unsupervised longitudinal, semi-supervised independent and semi-supervised longitudinal), evaluated with this new data distribution. Only Cosine-similarity Mean-Shift pseudo-speaker labels are considered.

Table 9.4: DER (%) results in the evaluation set with multiple adaptations of configuration: None, Independent or Longitudinal unsupervised adaptation and with or without previous supervised adaptation

Unsup. Adapt.	No Prev. Sup. Adapt	With Prev. Sup. Adapt
None	41.65	39.00
Ind. Adapt	**35.39**	**33.88**
Long. Adapt	37.00	35.68

According to Table 9.4, all our totally unsupervised approaches (independent and longitudinal), obtain some boost in performance by including a supervised adaptation step, becoming semi-supervised approaches. In fact, all the results without any supervised adaptation (the baseline and the totally unsupervised adaptations) are improved similarly (approximately 2% absolute improvement). Hence supervised and unsupervised adaptations are complementary.

9.4 Conclusions

The work done along this chapter provides a detailed analysis of domain adaptation as a solution for the problem of domain mismatch, noticeable in broadcast data. Different approaches based on supervised and specially unsupervised PLDA adaptations, including hybrid solutions, were tested. Our main goal is the validation of our novel unsupervised adaptation methods, which allow the substitution of manually obtained speaker labels by automatically obtained pseudo-speaker labels. This technology reduces the need for in-domain labeled data, with its respective reduction of expenses.

The most important result is that our novel unsupervised adaptation approaches are able to outperform a supervised adaptation when perfectly labeled in-domain data is scarce. Our results revealed up to 9% relative improvements when comparing the new totally unsupervised approaches versus a supervised adaptation. Therefore, in-domain information automatically estimated from the data we are diarizing can be more informative than small amounts of manually annotated in-domain data. Besides both adaptations, supervised and our unsupervised one, are totally compatible. The results indicate that improvements are accumulated if both adaptation approaches are applied. Our hybrid adaptations implied up to 13% relative improvement compared to considering only a supervised adaptation. All these improvements offer multiple opportunities. On the one hand the reduction of the need for manually labeled data is possible, partially substituting hand-transcribed data with unsupervised pseudo-speaker labels. On the other hand, this technique can offer a significant boost of performance by just making a more efficient use of the available data, including the evaluation audio itself.

Despite outperforming the baseline and the supervised adaptation, not all the proposed architectures performed similarly. The results show that those strategies which deal independently with the episodes (independent adaptation) obtained better results than considering all of them (our longitudinal approach). In the context of MGB 2015, the former obtained a relative 16% improvement while the latter got a relative 13% improvement with respect to the baseline. This general loss of performance with respect to the independent adaptation approach indicates that our proposed longitudinal adaptation takes no further advantage of automatically labeled in-domain data, being degraded by the accumulated errors. Further work should find strategies that successfully make use of this available extra information.

Finally, our results reassure that simple techniques such as AHC and MS are accurate enough to generate improvements working as initialization. However, not all these labels are equally useful. All our results indicate that MS performs significantly better than the AHC, and the cosine similarity pseudo-speaker labels outperform PLDA-based ones.

Part V

Conclusions & Future Work

Chapter 10

Conclusions & Future work

10.1 Conclusions

Along the different chapters of this thesis a deep study of the problem of diarization has been carried out, with special emphasis on the broadcast domain. Along the different experiments we have shown that broadcast data collected "in the wild" is a domain characterized by a large variability. In fact, we can observe a combination of local domains, the TV shows, each one with its own characteristics. The developed analysis has also illustrated that this variability influences the acoustic conditions as well as the speaker distribution. Furthermore, our study goes further, illustrating that show particular conditions may not be stable along the whole show, only remaining very correlated for limited periods of time, also referred along the thesis as sections of the show.

In addition to the analysis of broadcast data nature we have also made the effort for the improvement of three different diarization subtasks: the clustering stage, the speaker characterization task and the domain mismatch compensation

10.1.1 The clustering task

Our first part of the thesis works in terms of the FBPLDA model. This model has significantly improved our results in our two datasets of interest, MGB 2015 and Albayzín 2018. In the process this model has succeeded in reducing the variability about number of speakers compared with our AHC baseline. Nevertheless, its initialization limitations cannot guarantee the best possible result, thus multiple starting scenarios can be considered at the same time. In addition to the extra computational cost, this parallel strategy adds an extra complexity, the inclusion of a choice criterion. Fortunately, according to our experiments the use of the Evidence

Lower Bound (ELBO) and its penalized version behave as reasonable good indicators of the best obtained partition.

A proposed evolution of the FBPLDA model is the FBPLDAUP, a similar model now including the concept of uncertainty propagation, i.e., the consideration of the i-vector covariance. During our experiments in telephone channel uncertainty propagation has shown promising results when working with short utterances in both speaker recognition and specially speaker clustering. Similar results were obtained in speaker clustering in broadcast data. However, issues have appeared when FBPLDAUP resegmentation was applied on the broadcast domain. According to our results, the new proposed model successfully outperforms FBPLDA when a fixed number of initial speakers is *a priori* given. Nevertheless, the parallel strategy using ELBO as partition selector does not work well with FBPLDAUP, not obtaining any further improvement. Moreover, the current implementation of FBPLDAUP has demonstrated significant higher computational costs with respect to FBPLDA. Therefore, uncertainty propagation is an interesting concept to take into account for i-vectors and alternative embeddings, despite alternative ways to deal with the uncertainty are needed.

Finally, we move out the FBPLDA resegmentation paradigm, proposing a PLDA tree-based clustering strategy that exploits the product rule of probability. According to our experiments, this new technique is able to slightly outperform FBPLDA while reducing the intershow variability behaviour. Our experiments have also shown the relevance of the parameter M, i.e. the number of surviving paths in the M-algorithm, showing significant improvements as long as few paths are simultaneously evaluated. Moreover, this parameter is a powerful control about the search tradeoff, balancing computational costs and efficiency.

10.1.2 The speaker characterization stage

Our clustering research is complemented by our analysis of embedding extraction, which has led to interesting results. The work done has provided us a wider understanding about the embedding principle, how they store the information and why short utterances are more likely to fail. This understanding has crucial relevance in diarization, where the variability of short segments is usually attributed to speaker mismatches.

Taking into account the obtained results, representations for short utterances are specially weak in terms of robustness because the speech content is too specific and thus, having a low phonetic balance resemblance with respect to other utterances. Besides, according to our experiments target trials are significantly more affected by this balance mismatch between enrollment and test data, being more easily misclassified if this dissimilarity increases. Furthermore, we

have identified a range of mismatch where degradation performance is bounded, being highly degraded otherwise. Furthermore, our experiments indicate that once perfect match of the distributions is achieved, further information in extra components does not provide any significant improvement in performance.

The acquisition of this knowledge, despite obtained from i-vectors, can also be applied to DNN-based embeddings too. Thus, we have also analyzed some of the state-of-the-art embeddings, which make use themselves of our proposed knowledge, evaluating them in diarization. Regarding this evaluation we have analyzed the performance of two hybrid i-vectors strategies, one based on BNFs and another based on phonetic posteriors. Our evaluation is also expanded to purely DNN strategies as x-vectors. According to our experiments, BNFs have shown to be not so robust with broadcast data. However, we have shown the impact of the position of the bottleneck extraction layer within the DNN, obtaining more accurate results as long as the bottleneck is closer to the input of the network. The phonetic i-vector approach is our alternative embedding strategy to BNF i-vectors. This architecture takes advantage of a phoneme classifier to improve the responsibility classification. Our results evidence that more distinguishable acoustic units lead to more accurate results in both speaker recognition and diarization, even with short utterances. Regarding x-vectors, according to our results they specially outperform i-vectors in the short utterance scenario, even though DNNs were trained with limited data. When applied to diarization x-vectors clearly outperform i-vectors with our two types of clustering. Besides, this improvement is more noticeable with the proposed PLDA tree-based sequential architecture.

10.1.3 Unsupervised domain adaptation research

Our final line of research is the domain adaptation task, in low-resource situations. First of all, totally unsupervised and semi-supervised strategies have been tested, with satisfactory improvements. This success let us substitute costly manually obtained labels by automatically estimated ones.

When considering an adaptation approaches, the obtained results show that the knowledge acquired from the audio to diarize itself in an unsupervised manner can be worthy for adaptation purposes despite the estimation mistakes. In fact, according to our experiments this extracted knowledge is sometimes more valuable than manually transcribed data from the same domain. Moreover, both sources of knowledge are compatible, accumulating their gains.

With respect to the specific technique, we have tested two main adaptation strategies, independent and longitudinal. Both have demonstrated significant improvements compared to the

reference system, although gains are slightly different. The obtained results indicate that an individual treatment of each episode is more convenient for diarization purposes compared to its longitudinal counterpart. This latter approach does not achieve to longitudinally accumulate more knowledge along multiple adaptations, degrading the model instead.

Finally, our results reassure that simple techniques such as Agglomerative Hierarchical Clustering and Mean-Shift are accurate enough to generate improvements working as initialization. However, not all these labels are equally useful. All our results indicate that Mean-Shift performs significantly better than the Agglomerative Hierarchical Clustering, and the cosine similarity pseudo-speaker labels outperform PLDA-based ones.

10.2 Scientific Contributions

The research work carried out along this thesis produced multiple contributions to books, peer-review journals and conference proceedings, involving the three main lines of research in this thesis. These contributions are:

10.2.1 Book chapters

- **Speaker characterization**
  - I. Viñals, J. Villalba, A. Ortega, A. Miguel, E. Lleida, *Bottleneck based front-end for diarization systems*, International Conference on Advances in Speech and Language Technologies for Iberian Languages, 2016, 276-286.

10.2.2 Papers published in journals included in the Journal Citation Reports (JCR)

- **Speaker characterization**
  - I. Viñals, A. Ortega, A. Miguel, E. Lleida, *An Analysis of the Short Utterance Problem for Speaker Characterization*, Applied Sciences 9 (18) 3697, 2019.
- **Domain adaptation**
  - I. Viñals, A. Ortega, J. Villalba, A. Miguel, E. Lleida, *Unsupervised Adaptation of PLDA Models for Broadcast Diarization*, EURASIP Journal on Audio, Speech and Music Processing, 2019:24, 2019.

10.2.3 Conference proceedings

- **The clustering task**
  - I. Viñals, P. Gimeno, A. Ortega, A. Miguel, E. Lleida, *Estimation of the Number of Speakers with Variational Bayesian PLDA in the DIHARD Diarization Challenge*, Interspeech 2018. 2803-2807.
  - I. Viñals, P. Gimeno, A. Ortega, A. Miguel, E. Lleida, *ViVoLAB Speaker Diarization System for the DIHARD 2019 Challenge*, Interspeech, 2019, 988-992
- **Speaker characterization**
  - I. Viñals, A. Ortega, A. Miguel, E. Lleida, *Phonetic Variability Influence on Short Utterances in Speaker Verification*, International Conference on Advances in Speech and Language Technologies for Iberian Languages, 2018, 6-9
  - I. Viñals, D. Ribas, V. Mingote, J. Llombart, P. Gimeno, A. Miguel, A. Ortega, E. Lleida, *Phonetically-aware embeddings, Wide Residual Networks with Time-Delay Neural Networks and Self Attention models for the 2018 NIST Speaker Recognition Evaluation*, Interspeech, 2019, 4310-4314
- **Domain adaptation**
  - I. Viñals, A. Ortega, J. Villalba, A. Miguel, E. Lleida, *Domain Adaptation of PLDA Models in Broadcast Diarization by Means of Unsupervised Speaker Clustering*. Interspeech, 2017. 2829-2833.
  - I. Viñals, P. Gimeno, A. Ortega, A. Miguel, E. Lleida, *In-domain Adaptation Solutions for the RTVE 2018 Diarization Challenge*, International Conference on Advances in Speech and Language Technologies for Iberian Languages, 2018, 220-223.

10.3 Future Work

In these final lines we explore how to continue the lines of research followed along the thesis work. Due to the fact that this thesis has covered three different subtasks within the diarization problem, we will specifically provide an explanation for each one of them.

The clustering step is probably the task with more opportunities to work in. Considering the obtained results, the FBPLDA resegmentation strategy is a powerful approach with a great

limitation due to the initialization. Therefore, any obtained improvement in the initialization estimation should have an immediate benefit for the FBPLDA strategy. Considering the proposed parallel evaluation approach, a more efficient substitute for ELBO as a partition selector may be important too. Regarding efficiency, the estimation of the initialization quality before the simultaneous reclustering means a great reduction of computational costs.

Moving to the sequential clustering, its performance has demonstrated to be powerful, and the fact that prior initialization is not needed makes it very interesting for its posterior development. Regarding its evolutions, the prediction of the current decision based on past and future events may be the most promising option. Moreover, quality measures to determine how local decisions influence the analyzed paths are also viable.

Regarding the uncertainty propagation, this line of research seems *a priori* old-fashioned, specially considering that current embeddings are not estimated with any uncertainty measure. Nevertheless, some works such as [Brummer et al., 2017] propose the substitution of point estimations (embeddings) by statistical distributions (meta-embeddings). Hence, if any extra information was estimated, uncertainty propagation would become viable again. However, for this reason more efficient strategies to incorporate the uncertainty should be designed, specially bearing in mind computational resources.

Our contributions to the embedding extraction were more focused on the analytic part of the problem. Our analysis divided the most typical embedding extraction as the inference of a set of particular discriminative patterns, unsupervisedly learnt, followed by a weighted combination. While many improvements have been proposed with respect to the extraction of new patterns, no contribution has tried to balance the available information, inferring the missing patterns. This type of evolution might lead to a significant new level when considering short utterances.

Finally, the last part of the thesis was dedicated to model adaptation. Regarding this task, our improvements are very linked to the pseudo-speaker label estimation. The better these labels are, the better will be the posterior adaptation. However, these labels should be obtained by simple techniques, making the computational effort with the final diarization with more elaborated strategies. Another related task to improve is the longitudinal adaptation. This strategy was conceived for a scenario in which more and more data is available along time, hence any approach capable of unsupervisedly extracting new information and wisely adding to previous knowledge might provide a cutting-edge approach.

Part VI

Appendix

Appendix A

Fully Bayesian PLDA with Uncertainty Propagation

This appendix is dedicated to a more detailed description of the Fully Bayesian PLDA with Uncertainty Propagation, previously mentioned in Chapter 5. Its Bayesian network is shown in Fig. A.1

A.1 Definitions

$$\boldsymbol{\phi}_j = \boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \mathbf{U}_j\mathbf{x}_{ij} + \epsilon_j \tag{A.1}$$

$$\mathbf{y}_i \sim \mathcal{N}\left(\mathbf{y}_i|0, I\right) \tag{A.2}$$

$$\mathbf{x}_{ij} \sim \mathcal{N}\left(\mathbf{x}_{ij}|0, I\right) \tag{A.3}$$

$$\epsilon_j \sim \mathcal{N}\left(\epsilon_j|0, \mathbf{W}^{-1}\right) \tag{A.4}$$

$$\mathbf{U}_j\mathbf{U}_j^T = \mathbf{B}_j^{-1} \tag{A.5}$$

$$\mathbf{B}_j = \mathbf{I} + \sum_{c=1}^{C} \mathbf{V}_{FA_c}^T \mathbf{N}_{FA_c}(j) \Sigma_{FA_c} \mathbf{V}_{FA_c} \tag{A.6}$$

$$P\left(\Theta|\pi_\theta\right) = \prod_{i=1}^{I}\prod_{j=1}^{N} \pi_{\theta_i}^{\theta_{ij}} \tag{A.7}$$

$$P\left(\pi_\theta|\tau_0\right) = C(\tau_0) \prod_{i=1}^{I} \pi_{\theta_i}^{\tau_0 - 1} \tag{A.8}$$

$$C(\tau_0) = \frac{\Gamma(I\tau_0)}{\Gamma(\tau_0)^I} \tag{A.9}$$

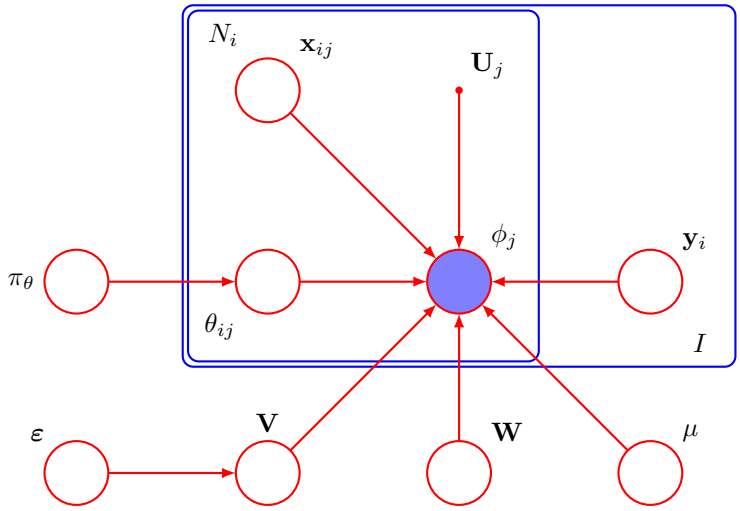


Figure A.1: Bayesian Network of the Fully Bayesian PLDA with Uncertainty Propagation

$$P\left(\mathbf{V}|\boldsymbol{\varepsilon}\right) = \prod_{q=1}^{n_y} \left(\frac{\boldsymbol{\varepsilon}_q}{2\pi}\right)^{\frac{d}{2}} \exp\left(-\frac{1}{2}\boldsymbol{\varepsilon}_q \mathbf{v}_q^T \mathbf{v}_q\right) \tag{A.10}$$

$$P\left(\boldsymbol{\mu}\right) = \mathcal{N}\left(\boldsymbol{\mu}|\boldsymbol{\mu}_0, \mathrm{diag}(\beta)^{-1}\right) \tag{A.11}$$

$$P\left(\boldsymbol{\varepsilon}\right) = \prod_{q=1}^{n_y} \mathcal{G}\left(\boldsymbol{\varepsilon}_q|a_{\boldsymbol{\varepsilon}}, b_{\boldsymbol{\varepsilon}}\right) \tag{A.12}$$

$$P\left(\mathbf{W}\right) = \lim_{k\to 0} \mathcal{W}\left(\mathbf{W}|\frac{\mathbf{W}_0}{k}, k\right) = \alpha|\mathbf{W}|^{-\frac{d+1}{2}} \tag{A.13}$$

A.2 Data

Prior to the calculations we first define the following terms accumulating the speaker information:

$$\mathrm{E}_\Theta\left[N_i\right] = \sum_{j=1}^{N} \mathrm{E}\left[\theta_{ij}\right] \tag{A.14}$$

$$\mathrm{E}_\Theta\left[N\right] = \sum_{i=1}^{I} \mathrm{E}_\Theta\left[N_i\right] \tag{A.15}$$

$$\mathrm{E}_\Theta\left[\mathbf{F}_i\right] = \sum_{j=1}^{N} \mathrm{E}\left[\theta_{ij}\right] \boldsymbol{\phi}_j \tag{A.16}$$

$$\mathrm{E}_\Theta\left[\overline{\mathbf{F}}_i\right] = \sum_{j=1}^{N} \mathrm{E}\left[\theta_{ij}\right] \left(\boldsymbol{\phi}_j - \boldsymbol{\mu}\right) \tag{A.17}$$

$$\mathrm{E}_{\Theta}\left[\mathbf{F}\right] = \sum_{i=1}^{I} \mathrm{E}_{\Theta}\left[\mathbf{F}_i\right] \tag{A.18}$$

$$\mathrm{E}_{\Theta}\left[\overline{\mathbf{F}}\right] = \sum_{i=1}^{I} \mathrm{E}_{\Theta}\left[\overline{\mathbf{F}}_i\right] \tag{A.19}$$

$$\mathrm{E}_{\Theta}\left[\mathbf{S}_i\right] = \sum_{j=1}^{N} \mathrm{E}\left[\theta_{ij}\right] \boldsymbol{\phi}_j \boldsymbol{\phi}_j^T \tag{A.20}$$

$$\mathrm{E}_{\Theta}\left[\overline{\mathbf{S}}_i\right] = \sum_{j=1}^{N} \mathrm{E}\left[\theta_{ij}\right] (\boldsymbol{\phi}_j - \boldsymbol{\mu})(\boldsymbol{\phi}_j - \boldsymbol{\mu})^T \tag{A.21}$$

$$\mathrm{E}_{\Theta}\left[\mathbf{S}\right] = \sum_{i=1}^{I} \mathrm{E}_{\Theta}\left[\mathbf{S}_i\right] \tag{A.22}$$

$$\mathrm{E}_{\Theta}\left[\overline{\mathbf{S}}\right] = \sum_{i=1}^{I} \mathrm{E}_{\Theta}\left[\overline{\mathbf{S}}_i\right] \tag{A.23}$$

A.3 Data conditional likelihood

A.3.1 $P\left(\boldsymbol{\Phi}_i|\mathbf{y}_i, \mathbf{X}_i, \Theta_i, \mathcal{M}\right)$

$$\ln P\left(\boldsymbol{\Phi}_i|\mathbf{y}_i, \mathbf{X}_i, \Theta_i, \mathcal{M}\right) = \sum_{j=1}^{N} \theta_{ij} \ln \mathcal{N}\left(\boldsymbol{\phi}_j|\boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \mathbf{U}_j\mathbf{x}_{ij}, \mathbf{W}^{-1}\right) \tag{A.24}$$

$$\begin{aligned} &= \sum_{j=1}^{N} \theta_{ij} \left[\frac{1}{2} \ln \left|\frac{\mathbf{W}}{2\pi}\right| - \frac{1}{2}\mathrm{tr}\left[\mathbf{W}\left((\boldsymbol{\phi}_j - \boldsymbol{\mu})(\boldsymbol{\phi}_j - \boldsymbol{\mu})^T + \mathbf{V}\mathbf{y}_i\mathbf{y}_i^T\mathbf{V}^T\right.\right.\right. \\ &\quad \left.\left.\left. - 2(\boldsymbol{\phi}_j - \boldsymbol{\mu})\left(\mathbf{x}_{ij}^T\mathbf{U}_j^T + \mathbf{y}_i^T\mathbf{V}^T\right) + 2\mathbf{U}_j\mathbf{x}_{ij}\mathbf{y}_i^T\mathbf{V}^T + \mathbf{U}_j\mathbf{x}_{ij}\mathbf{x}_{ij}^T\mathbf{U}_j^T\right)\right]\right] \end{aligned} \tag{A.25}$$

An alternative definition used along this appendix considers

$$\tilde{\mathbf{V}} = \begin{bmatrix}\mathbf{V} & \boldsymbol{\mu}\end{bmatrix}; \tilde{\mathbf{y}} = \begin{bmatrix}\mathbf{y} \\ 1\end{bmatrix} \tag{A.26}$$

Thus, the previous definition can also been written as:

$$\ln P\left(\boldsymbol{\Phi}_i|\tilde{\mathbf{y}}_i, \mathbf{X}_i, \Theta_i, \mathcal{M}\right) = \sum_{j=1}^{N} \theta_{ij} \ln \mathcal{N}\left(\boldsymbol{\phi}_j|\tilde{\mathbf{V}}\tilde{\mathbf{y}}_i + \mathbf{U}_j\mathbf{x}_{ij}, \mathbf{W}^{-1}\right) \tag{A.27}$$

$$\begin{aligned} &= \sum_{j=1}^{N} \theta_{ij} \left[\frac{1}{2} \ln \left|\frac{\mathbf{W}}{2\pi}\right| - \frac{1}{2}\mathrm{tr}\left[\mathbf{W}\left(\boldsymbol{\phi}_j\boldsymbol{\phi}_j^T + \tilde{\mathbf{V}}\tilde{\mathbf{y}}_i\tilde{\mathbf{y}}_i^T\tilde{\mathbf{V}}^T\right.\right.\right. \\ &\quad \left.\left.\left. - 2\boldsymbol{\phi}_j\left(\mathbf{x}_{ij}^T\mathbf{U}_j^T + \tilde{\mathbf{y}}_i^T\tilde{\mathbf{V}}^T\right) + 2\mathbf{U}_j\mathbf{x}_{ij}\tilde{\mathbf{y}}_i^T\tilde{\mathbf{V}}^T + \mathbf{U}_j\mathbf{x}_{ij}\mathbf{x}_{ij}^T\mathbf{U}_j^T\right)\right]\right] \end{aligned} \tag{A.28}$$

A.3.2 $P\left(\mathbf{X}_i|\mathbf{y}_i,\Theta_i,\mathbf{\Phi}_i,\mathcal{M}\right)$

By Bayes Theorem, we know:

$$P\left(\mathbf{X}_i|\mathbf{y}_i,\Theta_i,\mathbf{\Phi}_i,\mathcal{M}\right)=\frac{P\left(\mathbf{\Phi}_i|\mathbf{y}_i,\mathbf{X}_i,\Theta_i,\mathcal{M}\right)P\left(\mathbf{X}_i\right)}{P\left(\mathbf{\Phi}_i|\mathbf{y}_i,\Theta_i\right)} \tag{A.29}$$

Then

$$\ln P\left(\mathbf{X}_i|\mathbf{y}_i,\Theta_i,\mathbf{\Phi}_i,\mathcal{M}\right)=\ln P\left(\mathbf{\Phi}_i|\mathbf{y}_i,\mathbf{X}_i,\Theta_i,\mathcal{M}\right)+\ln P\left(\mathbf{X}_i\right)+\text{const} \tag{A.30}$$

$$=\sum_{j=1}^{N}\mathbf{x}_{ij}^T\mathbf{U}_j^T\mathbf{W}\theta_{ij}(\boldsymbol{\phi}_j-\boldsymbol{\mu}-\mathbf{V}\mathbf{y}_i)-\frac{1}{2}\mathbf{x}_{ij}^T[\mathbf{I}+\mathbf{U}_j^T\mathbf{W}\theta_{ij}\mathbf{U}_j]\mathbf{x}_{ij}+\text{const} \tag{A.31}$$

Therefore, $P\left(\mathbf{X}_i|\mathbf{y}_i,\Theta_i,\mathbf{\Phi}_i,\mathcal{M}\right)$ is Gaussian distributed.

$$P\left(\mathbf{X}_i|\mathbf{y}_i,\Theta_i,\mathbf{\Phi}_i,\mathcal{M}\right)=\prod_{j=1}^{N}\mathcal{N}\left(\mathbf{x}_{ij}|\boldsymbol{\mu}_{\mathbf{x}_{ij}},\Sigma_{\mathbf{x}_{ij}}\right)=\prod_{j=1}^{N}\mathcal{N}\left(\mathbf{x}_{ij}|\mathbf{L}_{\mathbf{x}_{ij}}^{-1}\gamma_{\mathbf{x}_{ij}},\mathbf{L}_{\mathbf{x}_{ij}}^{-1}\right) \tag{A.32}$$

$$\mathbf{L}_{\mathbf{x}_{ij}}=\mathbf{I}+\mathbf{U}_j^T\mathbf{W}\theta_{ij}\mathbf{U}_j \tag{A.33}$$

$$\gamma_{\mathbf{x}_{ij}}=\theta_{ij}\mathbf{U}_j^T\mathbf{W}(\boldsymbol{\phi}_j-\boldsymbol{\mu}-\mathbf{V}\mathbf{y}_i) \tag{A.34}$$

A.3.3 $P\left(\mathbf{y}_i|\mathbf{\Phi}_i,\Theta_i,\mathcal{M}\right)$

Bayes Theorem defines

$$P\left(\mathbf{y}_i|\mathbf{\Phi}_i,\Theta_i,\mathcal{M}\right)=\frac{P\left(\mathbf{\Phi}_i|\mathbf{y}_i,\Theta_i\right)P\left(\mathbf{y}_i\right)}{P\left(\mathbf{\Phi}_i\right)} \tag{A.35}$$

In order to obtain certain identities, we consider

$$P\left(\mathbf{\Phi}_i,\mathbf{X}_i|\mathbf{y}_i,\Theta_i\right)=P\left(\mathbf{\Phi}_i|\mathbf{y}_i,\mathbf{X}_i,\Theta_i\right)P\left(\mathbf{X}_i|\mathbf{y}_i,\Theta_i\right) \tag{A.36}$$

$$=P\left(\mathbf{X}_i|\mathbf{\Phi}_i,\mathbf{y}_i,\Theta_i\right)P\left(\mathbf{\Phi}_i|\mathbf{y}_i,\Theta_i\right) \tag{A.37}$$

$$=P\left(\mathbf{\Phi}_i|\mathbf{y}_i,\mathbf{X}_i,\Theta_i\right)P\left(\mathbf{X}_i\right) \tag{A.38}$$

$$=P\left(\mathbf{X}_i|\mathbf{\Phi}_i,\mathbf{y}_i,\Theta_i\right)P\left(\mathbf{\Phi}_i|\mathbf{y}_i,\Theta_i\right) \tag{A.39}$$

Introducing the obtained equality into our starting formula

$$P\left(\mathbf{y}_i|\mathbf{\Phi}_i,\Theta_i,\mathcal{M}\right)=\frac{P\left(\mathbf{\Phi}_i|\mathbf{y}_i,\mathbf{X}_i,\Theta_i\right)P\left(\mathbf{X}_i\right)P\left(\mathbf{y}_i\right)}{P\left(\mathbf{\Phi}_i\right)P\left(\mathbf{X}_i|\mathbf{\Phi}_i,\mathbf{y}_i,\Theta_i\right)} \tag{A.40}$$

Because $P\left(\mathbf{y}_i|\mathbf{\Phi}_i,\Theta_i,\mathcal{M}\right)$ does not depend on $\mathbf{X}_i$, we can evaluate our obtained result in any value for $\mathbf{X}_i$, e.g., 0. In this case:

$$\begin{aligned}
&\ln P\left(\mathbf{y}_i|\mathbf{\Phi}_i,\Theta_i\right)=\ln P\left(\mathbf{\Phi}_i|\mathbf{y}_i,\mathbf{X}_i,\Theta_i\right)+\ln P\left(\mathbf{X}_i\right)\\
&+\ln P\left(\mathbf{y}_i\right)-\ln P\left(\mathbf{X}_i|\mathbf{\Phi}_i,\mathbf{y}_i,\Theta_i\right)+\text{const}\,|_{\mathbf{X}_i=0} &\text{(A.41)}\\
=&\mathbf{y}_i^T\mathbf{V}^T\mathbf{W}\sum_{j=1}^{N}\theta_{ij}(\boldsymbol{\phi}_j-\boldsymbol{\mu})-\frac{1}{2}\mathbf{y}_i^T\mathbf{V}^T\mathbf{W}\sum_{j=1}^{N}\theta_{ij}\mathbf{V}\mathbf{y}_i-\frac{1}{2}\mathbf{y}_i^T\mathbf{y}_i\\
&+\frac{1}{2}\sum_{j=1}^{N}(\boldsymbol{\phi}_j-\boldsymbol{\mu}-\mathbf{V}\mathbf{y}_i)^T\theta_{ij}\mathbf{W}\mathbf{U}_j\mathbf{L}_{\mathbf{x}_{ij}}\mathbf{U}_j^T\mathbf{W}\theta_{ij}(\boldsymbol{\phi}_j-\boldsymbol{\mu}-\mathbf{V}\mathbf{y}_i)+\text{const} &\text{(A.42)}\\
=&\mathbf{y}_i^T\left[\mathbf{V}^T\mathbf{W}\sum_{j=1}^{N}\theta_{ij}(\boldsymbol{\phi}_j-\boldsymbol{\mu})-\mathbf{V}^T\mathbf{W}\sum_{j=1}^{N}\theta_{ij}\theta_{ij}\mathbf{U}_j\mathbf{L}_{\mathbf{x}_{ij}}\mathbf{U}_j^T\mathbf{W}(\boldsymbol{\phi}_j-\boldsymbol{\mu})\right]\\
&-\frac{1}{2}\mathbf{y}_i^T\left[\mathbf{I}+\mathbf{V}^T\mathbf{W}\sum_{j=1}^{N}\theta_{ij}\mathbf{V}-\mathbf{V}\mathbf{W}\sum_{j=1}^{N}\theta_{ij}\theta_{ij}\mathbf{U}_j\mathbf{L}_{\mathbf{x}_{ij}}\mathbf{U}_j^T\mathbf{W}\mathbf{V}\right]\mathbf{y}_i+\text{const} &\text{(A.43)}
\end{aligned}$$

The last result indicates $P\left(\mathbf{y}_i|\mathbf{\Phi}_i,\Theta_i,\mathcal{M}\right)$ is Gaussian distributed in such a way:

$$\begin{aligned}
&P\left(\mathbf{y}_i|\mathbf{\Phi}_i,\Theta_i,\mathcal{M}\right)=\mathcal{N}\left(\mathbf{y}_i|\boldsymbol{\mu}_{\mathbf{y}_i},\Sigma_{\mathbf{y}_i}\right)=\mathcal{N}\left(\mathbf{y}_i|\mathbf{L}_{\mathbf{y}_i}^{-1}\gamma_{\mathbf{y}_i},\mathbf{L}_{\mathbf{y}_i}^{-1}\right) &\text{(A.44)}\\
&\mathbf{L}_{\mathbf{y}_i}=\mathbf{I}+\mathbf{V}^T\left(\mathbf{W}\sum_{j=1}^{N}\theta_{ij}-\mathbf{W}\sum_{j=1}^{N}\theta_{ij}\theta_{ij}\mathbf{U}_j\mathbf{L}_{\mathbf{x}_{ij}}\mathbf{U}_j^T\mathbf{W}\right)\mathbf{V} &\text{(A.45)}\\
&\gamma_{\mathbf{y}_i}=\mathbf{V}^T\left(\mathbf{W}\sum_{j=1}^{N}\theta_{ij}(\boldsymbol{\phi}_j-\boldsymbol{\mu})-\mathbf{W}\sum_{j=1}^{N}\theta_{ij}\theta_{ij}\mathbf{U}_j\mathbf{L}_{\mathbf{x}_{ij}}\mathbf{U}_j^T\mathbf{W}(\boldsymbol{\phi}_j-\boldsymbol{\mu})\right) &\text{(A.46)}
\end{aligned}$$

A.4 Variational approach

A.4.1 Joint probability

The probability conformed by all the terms is equal to:

$$\begin{aligned}
&P\left(\mathbf{\Phi},\mathbf{Y},\mathbf{X},\Theta,\pi_\theta,\boldsymbol{\mu},\mathbf{V},\mathbf{W},\boldsymbol{\varepsilon}|\tau_0\right) &\text{(A.47)}\\
=&P\left(\mathbf{\Phi}|\mathbf{Y},\mathbf{X},\Theta,\boldsymbol{\mu},\mathbf{V},\mathbf{W}\right)P\left(\mathbf{Y}\right)P\left(\mathbf{X}\right)\times\\
&P\left(\Theta|\pi_\theta\right)P\left(\pi_\theta|\tau_0\right)P\left(\boldsymbol{\mu}\right)P\left(\mathbf{V}|\boldsymbol{\varepsilon}\right)P\left(\boldsymbol{\varepsilon}\right)P\left(\mathbf{W}\right) &\text{(A.48)}
\end{aligned}$$

A.4.2 Variational Bayes approximation

We decompose the posterior probability in a product of factors in the following way:

$$P\left(\mathbf{Y},\mathbf{X},\Theta,\pi_\theta,\boldsymbol{\mu},\mathbf{V},\mathbf{W},\boldsymbol{\varepsilon}|\boldsymbol{\Phi}\right)\approx q\left(\mathbf{Y},\mathbf{X}\right)q\left(\Theta\right)q\left(\pi_\theta\right)\prod_{r=1}^{d}q\left(\tilde{\mathbf{v}}_r'\right)q\left(\boldsymbol{\varepsilon}\right)q\left(\mathbf{W}\right) \tag{A.49}$$

A.4.3 Optimal definition of $q^*\left(\mathbf{Y},\mathbf{X}\right)$

$$\begin{aligned}
\ln q^*\left(\mathbf{Y},\mathbf{X}\right) =& \mathrm{E}_{\Theta,\pi_\theta,\tilde{\mathbf{V}},\mathbf{W},\boldsymbol{\varepsilon}}\left[\ln P\left(\boldsymbol{\Phi},\mathbf{Y},\mathbf{X},\Theta,\pi_\theta,\tilde{\mathbf{V}},\mathbf{W},\boldsymbol{\varepsilon}\right)\right] && \text{(A.50)}\\
=& \mathrm{E}_{\Theta,\tilde{\mathbf{V}},\mathbf{W}}\left[\ln P\left(\boldsymbol{\Phi},\mathbf{Y},\mathbf{X}|\Theta,\tilde{\mathbf{V}},\mathbf{W}\right)\right]+\text{const} && \text{(A.51)}\\
=& \mathrm{E}_{\Theta,\tilde{\mathbf{V}},\mathbf{W}}\left[\ln P\left(\mathbf{Y}|\boldsymbol{\Phi},\Theta,\tilde{\mathbf{V}},\mathbf{W}\right)+\ln P\left(\mathbf{X}|\mathbf{Y},\boldsymbol{\Phi},\Theta,\tilde{\mathbf{V}},\mathbf{W}\right)\right]+\text{const} && \text{(A.52)}\\
=& \mathrm{E}_{\Theta,\tilde{\mathbf{V}},\mathbf{W}}\left[\sum_{i=1}^{I}\ln\mathcal{N}\left(\mathbf{y}_i|\boldsymbol{\mu}_{\mathbf{y}_i},\mathbf{L}_{\mathbf{y}_i}\right)+\sum_{i=1}^{I}\sum_{j=1}^{N}\ln\mathcal{N}\left(\mathbf{x}_{ij}|\boldsymbol{\mu}_{\mathbf{x}_{ij}},\mathbf{L}_{\mathbf{x}_{ij}}\right)\right]+\text{const} && \text{(A.53)}\\
=& \sum_{i=1}^{I}\ln\mathcal{N}\left(\mathbf{y}_i|\boldsymbol{\mu}_{\mathbf{y}_i},\mathbf{L}_{\mathbf{y}_i}\right)+\\
&+\sum_{i=1}^{I}\sum_{j=1}^{N}\ln\mathcal{N}\left(\mathbf{x}_{ij}|\boldsymbol{\mu}_{\mathbf{x}_{ij}},\mathbf{L}_{\mathbf{x}_{ij}}\right)+\text{const} && \text{(A.54)}\\
=& \ln(P_q(\mathbf{Y})P_q(\mathbf{X}|\mathbf{Y})) && \text{(A.55)}
\end{aligned}$$

so

$$\begin{aligned}
P_q(\mathbf{Y}) =& \prod_{i=1}^{I}\mathcal{N}\left(\mathbf{y}_i|\overline{\mathbf{y}}_i,\mathbf{L}_{\mathbf{y}_i}^{-1}\right) && \text{(A.56)}\\
\overline{\mathbf{y}}_i =& \mathbf{L}_{\mathbf{y}_i}^{-1}\sum_{j=1}^{N}\mathrm{E}\left[\mathbf{V}^T\theta_{ij}\left(\mathbf{W}^{-1}+\mathbf{U}_j\mathbf{U}_j^T\right)^{-1}\left(\boldsymbol{\phi}_j-\boldsymbol{\mu}\right)\right] && \text{(A.57)}\\
=& \mathbf{L}_{\mathbf{y}_i}^{-1}\left(\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\right]\mathrm{E}\left[\mathbf{V}^T\left(\mathbf{W}^{-1}+\mathbf{U}_j\mathbf{U}_j^T\right)^{-1}\right]\boldsymbol{\phi}_j\right.\\
&\left.-\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\right]\mathrm{E}\left[\mathbf{V}^T\left(\mathbf{W}^{-1}+\mathbf{U}_j\mathbf{U}_j^T\right)^{-1}\boldsymbol{\mu}\right]\right) && \text{(A.58)}\\
=& \mathbf{L}_{\mathbf{y}_i}^{-1}\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\right]\left(\mathrm{E}\left[\mathbf{V}\right]^T\mathrm{E}\left[\mathbf{B}_j\right]\boldsymbol{\phi}_j-\sum_{j=1}^{N}\mathrm{E}\left[\mathbf{V}^T\mathbf{B}_j\boldsymbol{\mu}\right]\right) && \text{(A.59)}
\end{aligned}$$

$$\mathbf{L}_{\mathbf{y}_i} = \mathbf{I} + \mathrm{E}\left[\sum_{j=1}^{N} \theta_{ij} \mathbf{V}^T \left(\mathbf{W}^{-1} + \mathbf{U}_j \mathbf{U}_j^T\right)^{-1} \mathbf{V}\right] \tag{A.60}$$

$$= \mathbf{I} + \sum_{j=1}^{N} \mathrm{E}\left[\theta_{ij}\right] \mathrm{E}\left[\mathbf{V}^T \left(\mathbf{W}^{-1} + \mathbf{U}_j \mathbf{U}_j^T\right)^{-1} \mathbf{V}\right] \tag{A.61}$$

$$= \mathbf{I} + \sum_{j=1}^{N} \mathrm{E}\left[\theta_{ij}\right] \mathrm{E}\left[\mathbf{V}^T \mathbf{B}_j \mathbf{V}\right] \tag{A.62}$$

and

$$P_q(\mathbf{X}|\mathbf{Y}) = \prod_{i=1}^{I} \prod_{j=1}^{N} \mathcal{N}\left(\mathbf{x}_{ij} | \overline{\mathbf{x}}_{ij}, \mathbf{L}_{\mathbf{x}_{ij}}^{-1}\right) \tag{A.63}$$

$$\overline{\mathbf{x}}_{ij} = \mathbf{L}_{\mathbf{x}_{ij}}^{-1} \mathrm{E}\left[\theta_{ij}\right] \mathbf{U}_j^T \left(\mathrm{E}\left[\mathbf{W}\right] \boldsymbol{\phi}_j - \mathrm{E}\left[\mathbf{W}\right] \mathrm{E}\left[\boldsymbol{\mu}\right] - \mathrm{E}\left[\mathbf{W}\right] \mathrm{E}\left[\mathbf{V}\right] \mathrm{E}\left[\mathbf{y}_i\right]\right) \tag{A.64}$$

$$\mathbf{L}_{\mathbf{x}_{ij}} = \mathbf{I} + \mathrm{E}\left[\mathbf{U}_j^T \mathbf{W} \theta_{ij} \mathbf{U}_j\right] = \mathbf{I} + \mathbf{U}_j^T \mathrm{E}\left[\mathbf{W}\right] \mathrm{E}\left[\theta_{ij}\right] \mathbf{U}_j \tag{A.65}$$

A.4.4 Optimal definition of $q^*(\Theta)$

Regarding $q^*(\Theta)$

$$\ln q^*(\Theta) = \mathrm{E}_{\mathbf{Y},\mathbf{X},\pi_\theta,\tilde{\mathbf{V}},\mathbf{W},\boldsymbol{\varepsilon}}\left[\ln P\left(\boldsymbol{\Phi}, \mathbf{Y}, \mathbf{X}, \Theta, \pi_\theta, \tilde{\mathbf{V}}, \mathbf{W}, \boldsymbol{\varepsilon}\right)\right] \tag{A.66}$$

$$= \mathrm{E}_{\mathbf{Y},\tilde{\mathbf{V}},\mathbf{W}}\left[\ln P\left(\boldsymbol{\Phi}|\mathbf{Y}, \Theta, \tilde{\mathbf{V}}, \mathbf{W}\right)\right] + \mathrm{E}_{\pi_\theta}\left[\ln P\left(\Theta|\pi_\theta\right)\right] + \text{const} \tag{A.67}$$

$$= \sum_{i=1}^{I} \sum_{j=1}^{N} \theta_{ij} \left[\mathrm{E}_{\pi_\theta}\left[\ln \pi_{\theta_i}\right] + \frac{1}{2}\mathrm{E}\left[\ln |\mathbf{B}_j|\right] - \frac{d}{2}\ln(2\pi)\right.$$
$$\left. - \frac{1}{2}\mathrm{E}\left[(\boldsymbol{\phi}_j - \tilde{\mathbf{V}}\tilde{\mathbf{y}})^T \mathbf{B}_j (\boldsymbol{\phi}_j - \tilde{\mathbf{V}}\tilde{\mathbf{y}})\right]\right] + \text{const} \tag{A.68}$$

Taking the exponential in both sides of the equality

$$q^*(\Theta) = \prod_{i=1}^{I} \prod_{j=1}^{N} r_{ij}^{\theta_{ij}} \tag{A.69}$$

being

$$r_{ij} = \frac{\varrho_{ij}}{\sum_{i=1}^{I} \varrho_{ij}} \tag{A.70}$$

and

$$\ln \varrho_{ij} = \frac{1}{2}\mathrm{E}\left[\ln(|\mathbf{B}_j|)\right] - \frac{d}{2}\ln(2\pi) - \frac{1}{2}\mathrm{E}\left[(\boldsymbol{\phi}_j - \tilde{\mathbf{V}}\tilde{\mathbf{y}})^T \mathbf{B}_j (\boldsymbol{\phi}_j - \tilde{\mathbf{V}}\tilde{\mathbf{y}})\right] + \mathrm{E}_{\pi_\theta}\left[\ln \pi_{\theta_i}\right] \tag{A.71}$$

A.4.5 Optimal definition of $q^*(\pi_\theta)$

The optimum for $q^*(\pi_\theta)$ is:

$$\begin{aligned}
\ln q^*(\pi_\theta) =& \mathrm{E}_{\mathbf{Y},\mathbf{X},\Theta,\tilde{\mathbf{V}},\mathbf{W},\boldsymbol{\varepsilon}}\left[\ln P\left(\boldsymbol{\Phi},\mathbf{Y},\mathbf{X},\Theta,\pi_\theta,\tilde{\mathbf{V}},\mathbf{W},\boldsymbol{\varepsilon}\right)\right] && \text{(A.72)}\\
=& \mathrm{E}_\Theta\left[\ln P\left(\Theta|\pi_\theta\right)\right] + \ln P\left(\pi_\theta|\tau_0\right) + \text{const} && \text{(A.73)}\\
=& \sum_{i=1}^{I}\sum_{j=1}^{N} \mathrm{E}\left[\theta_{ij}\right]\ln\pi_{\theta_i} + (\tau_0 - 1)\sum_{i=1}^{I}\ln\pi_{\theta_i} + \text{const} && \text{(A.74)}\\
=& \sum_{i=1}^{I}\left(\mathrm{E}_\Theta\left[N_i\right] + \tau_0 - 1\right)\ln\pi_{\theta_i} && \text{(A.75)}
\end{aligned}$$

so

$$\begin{aligned}
q^*(\pi_\theta) =& C(\tau)\prod_{i=1}^{I}\pi_{\theta_i}^{\tau_i - 1} && \text{(A.76)}\\
\tau_i =& \mathrm{E}_\Theta\left[N_i\right] + \tau_0 && \text{(A.77)}\\
C(\tau) =& \frac{\Gamma\left(\sum_{i=1}^{I}\tau_i\right)}{\prod_{i=1}^{I}\Gamma(\tau_i)} && \text{(A.78)}
\end{aligned}$$

A.4.6 optimal definition of $q^*\left(\tilde{\mathbf{V}}\right)$

The optimum factor $q^*\left(\tilde{\mathbf{v}}'_r\right)$ can be defined as:

$$\begin{aligned}
\ln q^*\left(\tilde{\mathbf{v}}'_r\right) =& \mathrm{E}_{\mathbf{Y},\Theta,\mathbf{W},\boldsymbol{\varepsilon},\tilde{\mathbf{v}}'_{s\neq r}}\left[\ln P\left(\boldsymbol{\Phi},\mathbf{Y},\Theta,\boldsymbol{\mu},\mathbf{V},\mathbf{W},\boldsymbol{\varepsilon}\right)\right] + \text{const} && \text{(A.79)}\\
=& \mathrm{E}_{\mathbf{Y},\Theta,\mathbf{W},\tilde{\mathbf{v}}'_{s\neq r}}\left[\ln P\left(\boldsymbol{\Phi}|\mathbf{Y},\Theta,\boldsymbol{\mu},\mathbf{V},\mathbf{W}\right)\right] + \mathrm{E}_{\boldsymbol{\varepsilon},\tilde{\mathbf{v}}'_{s\neq r}}\left[\ln P\left(\mathbf{V}|\boldsymbol{\varepsilon}\right)\right] + \text{const} && \text{(A.80)}\\
=& -\frac{1}{2}\mathrm{tr}\left(-2\mathrm{E}\left[\mathbf{W}\right]\left(\sum_{i=1}^{I}\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\right]\left(\boldsymbol{\phi}_j - \mathbf{U}_j\mathbf{x}_{ij}\right)\mathrm{E}\left[\tilde{\mathbf{y}}_i\right]^T\mathrm{E}_{\tilde{\mathbf{v}}'_{s\neq r}}\left[\tilde{\mathbf{V}}\right]^T\right.\right.\\
&\left.\left. + \sum_{i=1}^{I}\mathrm{E}_{\Theta,\tilde{\mathbf{v}}'_{s\neq r}}\left[\tilde{\mathbf{V}}N_i\tilde{\mathbf{y}}_i\tilde{\mathbf{y}}_i^T\tilde{\mathbf{V}}^T\right]\right)\right)\\
&- \frac{1}{2}\sum_{q=1}^{n_y}\mathrm{E}\left[\boldsymbol{\varepsilon}_q\right]\mathrm{E}_{\tilde{\mathbf{v}}'_{s\neq r}}\left[\mathbf{v}_q^T\mathbf{v}_q\right] - \frac{1}{2}\beta_r(\boldsymbol{\mu}_r - \boldsymbol{\mu}_{0_r})^2 + \text{const} && \text{(A.81)}\\
=& -\frac{1}{2}\mathrm{tr}\left(-2\mathbf{A}\mathrm{E}_{\tilde{\mathbf{v}}'_{s\neq r}}\left[\mathbf{V}^T\right] + \mathrm{E}\left[\mathbf{W}\right]\mathrm{E}_{\tilde{\mathbf{v}}'_{s\neq r}}\left[\tilde{\mathbf{V}}\mathbf{R}_{\tilde{\mathbf{y}}}\tilde{\mathbf{V}}\right]\right)\\
&- \frac{1}{2}\sum_{q=1}^{n_y}\mathrm{E}\left[\boldsymbol{\varepsilon}_q\right]\mathrm{E}_{\tilde{\mathbf{v}}'_{s\neq r}}\left[\mathbf{v}_q^T\mathbf{v}_q\right] - \frac{1}{2}\beta_r(\boldsymbol{\mu}_r - \boldsymbol{\mu}_{0_r})^2 + \text{const} && \text{(A.82)}
\end{aligned}$$

$$= -\frac{1}{2}\text{tr}\left(-2\text{E}_{\tilde{\mathbf{v}}'_{s\neq r}}\left[\mathbf{V}^T\right]\mathbf{A} + \text{E}_{\tilde{\mathbf{v}}'_{s\neq r}}\left[\tilde{\mathbf{V}}\mathbf{W}\tilde{\mathbf{V}}\right]\mathbf{R}_{\tilde{\mathbf{y}}}\right)$$
$$-\frac{1}{2}\mathbf{v}_r'^T\text{diag}\left(\text{E}\left[\boldsymbol{\varepsilon}\right]\right)\mathbf{v}'_r - \frac{1}{2}\beta_r(\boldsymbol{\mu}_r - \boldsymbol{\mu}_{0_r})^2 + \text{const} \tag{A.83}$$

$$= -\frac{1}{2}\text{tr}\left(-2\sum_{s=1}^{d}\tilde{\mathbf{v}}'_r\boldsymbol{\rho}_r + 2\sum_{s\neq r}\tilde{\mathbf{v}}'_r\nu_{rs}\text{E}\left[\tilde{\mathbf{v}}'_s\right]^T\mathbf{R}_{\tilde{\mathbf{y}}} + \tilde{\mathbf{v}}'_r\nu_{rr}\tilde{\mathbf{v}}_r'^T\mathbf{R}_{\tilde{\mathbf{y}}}\right)$$
$$-\frac{1}{2}\mathbf{v}_r'^T\text{diag}\left(\text{E}\left[\boldsymbol{\varepsilon}\right]\right)\mathbf{v}'_r - \frac{1}{2}\beta_r(\boldsymbol{\mu}_r - \boldsymbol{\mu}_{0_r})^2 + \text{const} \tag{A.84}$$

$$= -\frac{1}{2}\text{tr}\left(-2\tilde{\mathbf{v}}'_r\left(\boldsymbol{\rho}_r - \sum_{s\neq r}\nu_{rs}\text{E}\left[\tilde{\mathbf{v}}'_s\right]^T\mathbf{R}_{\tilde{\mathbf{y}}}\right) + \tilde{\mathbf{v}}'_r\tilde{\mathbf{v}}_r'^T\nu_{rr}\mathbf{R}_{\tilde{\mathbf{y}}}\right)$$
$$-\frac{1}{2}\mathbf{v}_r'^T\text{diag}\left(\text{E}\left[\boldsymbol{\varepsilon}\right]\right)\mathbf{v}'_r - \frac{1}{2}\beta_r(\boldsymbol{\mu}_r - \boldsymbol{\mu}_{0_r})^2 + \text{const} \tag{A.85}$$

$$= -\frac{1}{2}\text{tr}\left(-2\tilde{\mathbf{v}}'_r\left(\boldsymbol{\rho}_r - \sum_{s\neq r}\nu_{rs}\text{E}\left[\tilde{\mathbf{v}}'_s\right]^T\mathbf{R}_{\tilde{\mathbf{y}}}\right) + \tilde{\mathbf{v}}'_r\tilde{\mathbf{v}}_r'^T\nu_{rr}\mathbf{R}_{\tilde{\mathbf{y}}}\right)$$
$$-\frac{1}{2}\tilde{\mathbf{v}}_r'^T\text{diag}\left(\overline{\tilde{\boldsymbol{\alpha}}}\right)\tilde{\mathbf{v}}'_r + \beta_r\boldsymbol{\mu}_r\boldsymbol{\mu}_{0_r} + \text{const} \tag{A.86}$$

$$= -\frac{1}{2}\text{tr}\left(-2\tilde{\mathbf{v}}'_r\left(\boldsymbol{\rho}_r - \sum_{s\neq r}\nu_{rs}\text{E}\left[\tilde{\mathbf{v}}'_s\right]^T\mathbf{R}_{\tilde{\mathbf{y}}} + \beta_r\tilde{\boldsymbol{\mu}}_{0_r}^T\right)\right.$$
$$\left.+\tilde{\mathbf{v}}'_r\tilde{\mathbf{v}}_r'^T\left(\text{diag}\left(\overline{\tilde{\boldsymbol{\alpha}}}_r\right) + \nu_{rr}\mathbf{R}_{\tilde{\mathbf{y}}}\right)\right) + \text{const} \tag{A.87}$$

where $\boldsymbol{\rho}_r$ is the $\mathbf{A}$ rth row and ν_{rs} is the element r, s in $\mathbf{W}$,

$$\mathbf{A} = \text{E}\left[\mathbf{W}\right]\left(\mathbf{C}_{\tilde{\mathbf{y}}} - \mathbf{C}_{\mathbf{x}\tilde{\mathbf{y}}}\right) \tag{A.88}$$

$$\mathbf{C}_{\tilde{\mathbf{y}}} = \sum_{i=1}^{I}\sum_{j=1}^{N}\text{E}\left[\theta_{ij}\right]\boldsymbol{\phi}_j\text{E}\left[\tilde{\mathbf{y}}_i\right]^T \tag{A.89}$$

$$\mathbf{C}_{\mathbf{x}\tilde{\mathbf{y}}} = \sum_{i=1}^{I}\sum_{j=1}^{N}\text{E}\left[\theta_{ij}\right]\mathbf{U}_j\text{E}\left[\mathbf{x}_{ij}\tilde{\mathbf{y}}_i^T\right] \tag{A.90}$$

$$\mathbf{R}_{\mathbf{y}} = \sum_{i=1}^{I}\text{E}_{\Theta}\left[N_i\right]\text{E}_{\mathbf{Y}}\left[\tilde{\mathbf{y}}_i\tilde{\mathbf{y}}_i^T\right] \tag{A.91}$$

Assuming the following expectations

$$\overline{\tilde{\boldsymbol{\alpha}}}_r = \begin{bmatrix}\text{E}\left[\boldsymbol{\varepsilon}\right]\\ \beta_r\end{bmatrix} \tag{A.92}$$

$$\tilde{\boldsymbol{\mu}}_{0_r} = \begin{bmatrix}0_{n_y\times 1}\\ \boldsymbol{\mu}_{0_r}\end{bmatrix} \tag{A.93}$$

In consequence, $q^*\left(\tilde{\mathbf{v}}'_r\right)$ is Gaussian distributed as:

$$q^*\left(\tilde{\mathbf{v}}'_r\right) = \mathcal{N}\left(\tilde{\mathbf{v}}'_r | \overline{\tilde{\mathbf{v}}}'_r, \mathbf{L}^{-1}_{\tilde{\mathbf{V}}_r}\right) \tag{A.94}$$

$$\mathbf{L}_{\tilde{\mathbf{V}}_r} = \text{diag}\left(\overline{\tilde{\boldsymbol{\alpha}}}_r\right) + \nu_{rr}\mathbf{R}_{\tilde{\mathbf{y}}} \tag{A.95}$$

$$\overline{\tilde{\mathbf{v}}}'_r = \mathbf{L}^{-1}_{\tilde{\mathbf{V}}_r}\left(\boldsymbol{\rho}_r^T - \sum_{s\neq r}\nu_{rs}\mathbf{R}_{\tilde{\mathbf{y}}}\overline{\tilde{\mathbf{v}}}'_s + \beta_r\tilde{\boldsymbol{\mu}}_{0_r}\right) \tag{A.96}$$

A.4.7 Optimal definition of $q^*\left(\mathbf{W}\right)$

Regarding $q^*\left(\mathbf{W}\right)$:

$$\ln q^*\left(\mathbf{W}\right) = \mathrm{E}_{\mathbf{Y},\Theta,\mathbf{X},\tilde{\mathbf{V}},\boldsymbol{\varepsilon}}\left[\ln P\left(\boldsymbol{\Phi},\mathbf{Y},\Theta,\mathbf{X},\boldsymbol{\mu},\mathbf{V},\mathbf{W}\right)\right] + \text{const} \tag{A.97}$$

$$= \mathrm{E}_{\mathbf{Y},\Theta,\mathbf{X},\tilde{\mathbf{V}},\boldsymbol{\varepsilon}}\left[\ln P\left(\boldsymbol{\Phi}|\mathbf{Y},\Theta,\mathbf{X},\boldsymbol{\mu},\mathbf{V},\mathbf{W}\right)\right] + \ln P\left(\mathbf{W}\right) + \text{const} \tag{A.98}$$

$$\begin{aligned} = &\frac{\xi_0 - d - 1}{2}\ln|\mathbf{W}| - \frac{1}{2}\text{tr}\left(\mathbf{W}\boldsymbol{\Psi}_0^{-1}\right) + \frac{1}{2}\sum_{i=1}^{I}\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\right]\ln\left|\frac{\mathbf{W}}{2\pi}\right| \\ &- \frac{1}{2}\text{tr}\left(\mathbf{W}\sum_{i=1}^{I}\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\left(\boldsymbol{\phi}_j - \tilde{\mathbf{V}}\tilde{\mathbf{y}}_i - \mathbf{U}_j\mathbf{x}_{ij}\right)\left(\boldsymbol{\phi}_j - \tilde{\mathbf{V}}\tilde{\mathbf{y}}_i - \mathbf{U}_j\mathbf{x}_{ij}\right)\right]\right) + \text{const} \end{aligned} \tag{A.99}$$

$$= \frac{\mathrm{E}_\Theta\left[N\right] + \xi_0 - d - 1}{2}\ln|\mathbf{W}| - \frac{1}{2}\text{tr}\left(\mathbf{W}\left(\boldsymbol{\Psi}_0^{-1} + \mathbf{K}\right)\right) + \text{const} \tag{A.100}$$

being

$$\mathbf{K} = \sum_{i=1}^{I}\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\left(\boldsymbol{\phi}_j - \tilde{\mathbf{V}}\tilde{\mathbf{y}}_i - \mathbf{U}_j\mathbf{x}_{ij}\right)\left(\boldsymbol{\phi}_j - \tilde{\mathbf{V}}\tilde{\mathbf{y}}_i - \mathbf{U}_j\mathbf{x}_{ij}\right)\right] \tag{A.101}$$

$$\begin{aligned} = &\mathrm{E}_\Theta\left[\mathbf{S}\right] - 2\mathrm{E}\left[\tilde{\mathbf{V}}\right]^T\sum_{i=1}^{I}\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\right]\left(\boldsymbol{\phi}_j\mathrm{E}\left[\tilde{\mathbf{y}}_i\right]^T - \mathbf{U}_j\mathrm{E}\left[\mathbf{x}_{ij}\tilde{\mathbf{y}}_i^T\right]\right) + \mathrm{E}\left[\tilde{\mathbf{V}}\mathbf{R}_{\tilde{\mathbf{y}}}\tilde{\mathbf{V}}^T\right] \\ &- 2\sum_{i=1}^{I}\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\right]\boldsymbol{\phi}_j\mathrm{E}\left[\mathbf{x}_{ij}\right]^T\mathbf{U}_j^T + \sum_{i=1}^{I}\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\right]\mathbf{U}_j\mathrm{E}\left[\mathbf{x}_{ij}\mathbf{x}_{ij}^T\right]\mathbf{U}_j^T \end{aligned} \tag{A.102}$$

$$= \mathrm{E}_\Theta\left[\mathbf{S}\right] - 2\left(\mathbf{C}_{\tilde{\mathbf{y}}} - \mathbf{C}_{\mathbf{x},\tilde{\mathbf{y}}}\right)\mathrm{E}\left[\tilde{\mathbf{V}}\right]^T + \mathrm{E}\left[\tilde{\mathbf{V}}\mathbf{R}_{\tilde{\mathbf{y}}}\tilde{\mathbf{V}}^T\right] - 2\mathbf{C}_{\mathbf{x}} + \mathbf{R}_{\mathbf{x}} \tag{A.103}$$

where

$$\mathbf{C}_{\mathbf{x}} = \sum_{i=1}^{I}\sum_{j=1}^{N}\mathrm{E}\left[\theta_{ij}\right]\boldsymbol{\phi}_j\mathrm{E}\left[\mathbf{x}_{ij}\right]^T\mathbf{U}_j^T \tag{A.104}$$

$$\mathbf{R}_{\mathbf{x}} = \sum_{i=1}^{I} \sum_{j=1}^{N} \mathrm{E}\left[\theta_{ij}\right] \mathbf{U}_j \mathrm{E}_{\mathbf{X}}\left[\mathbf{x}_{ij}\mathbf{x}_{ij}^T\right] \mathbf{U}_j^T \tag{A.105}$$

Thus $q^*\left(\mathbf{W}\right)$ can be interpreted as Wishart distributed:

$$P\left(\mathbf{W}\right) = \mathcal{W}\left(\mathbf{W}|\mathbf{\Psi}, \xi\right) \text{ if } \xi > d \tag{A.106}$$

$$\mathbf{\Psi}^{-1} = \mathbf{\Psi}_0^{-1} + \mathbf{K} \tag{A.107}$$

$$\xi = \xi_0 + N \tag{A.108}$$

A.4.8 Optimal definition of $q^*\left(\boldsymbol{\varepsilon}\right)$

Finally, the optimum for $q^*\left(\boldsymbol{\varepsilon}\right)$ is:

$$\ln q^*\left(\boldsymbol{\varepsilon}\right) = \mathrm{E}_{\mathbf{Y},\tilde{\mathbf{V}},\mathbf{W}}\left[\ln P\left(\mathbf{\Phi}, \mathbf{Y}, \boldsymbol{\mu}, \mathbf{V}, \mathbf{W}, \boldsymbol{\varepsilon}\right)\right] + \text{const} \tag{A.109}$$

$$= \mathrm{E}_{\mathbf{V}}\left[\ln P\left(\mathbf{V}|\boldsymbol{\varepsilon}\right)\right] + \ln P\left(\boldsymbol{\varepsilon}|a_{\boldsymbol{\varepsilon}}, b_{\boldsymbol{\varepsilon}}\right) + \text{const} \tag{A.110}$$

$$= \sum_{q=1}^{n_y} \frac{d}{2} \ln \boldsymbol{\varepsilon}_q - \frac{1}{2}\boldsymbol{\varepsilon}_q \mathrm{E}\left[\mathbf{v}_q^T \mathbf{v}_q\right] + \left(a_{\boldsymbol{\varepsilon}} - 1\right) \ln \boldsymbol{\varepsilon}_q - b_{\boldsymbol{\varepsilon}} \boldsymbol{\varepsilon}_q + \text{const} \tag{A.111}$$

$$= \sum_{q=1}^{ny} \left(\frac{d}{2} + a_{\boldsymbol{\varepsilon}} - 1\right) \ln \boldsymbol{\varepsilon}_q - \boldsymbol{\varepsilon}_q \left(b_{\boldsymbol{\varepsilon}} + \frac{1}{2}\mathrm{E}\left[\mathbf{v}_q^T \mathbf{v}_q\right]\right) + \text{const} \tag{A.112}$$

In consequence $q^*\left(\boldsymbol{\varepsilon}\right)$ can be interpreted as a product of Gamma distributions:

$$q^*\left(\boldsymbol{\varepsilon}\right) = \prod_{q=1}^{n_y} \mathcal{G}\left(\boldsymbol{\varepsilon}_q | a'_{\boldsymbol{\varepsilon}}, b'_{\boldsymbol{\varepsilon}_q}\right) \tag{A.113}$$

$$a'_{\boldsymbol{\varepsilon}} = a_{\boldsymbol{\varepsilon}} + \frac{d}{2} \tag{A.114}$$

$$b'_{\boldsymbol{\varepsilon}_q} = b_{\boldsymbol{\varepsilon}} + \frac{1}{2}\mathrm{E}\left[\mathbf{v}_q^T \mathbf{v}_q\right] \tag{A.115}$$

A.4.9 Necessary Expectations

$$\mathrm{E}\left[\mathbf{y}_i\right] = \overline{\mathbf{y}}_i \tag{A.116}$$

$$\mathrm{E}_{\mathbf{Y}}\left[\mathbf{y}_i \mathbf{y}_i^T\right] = \mathbf{L}_{\mathbf{y}_i}^{-1} + \overline{\mathbf{y}}_i \overline{\mathbf{y}}_i^T \tag{A.117}$$

$$\mathrm{E}\left[\tilde{\mathbf{y}}_i\right] = \begin{bmatrix} \mathrm{E}\left[\mathbf{y}_i\right] \\ 1 \end{bmatrix} \tag{A.118}$$

$$\mathrm{E}\left[\tilde{\mathbf{y}}_i \tilde{\mathbf{y}}_i^T\right] = \begin{bmatrix} \mathrm{E}_{\mathbf{Y}}\left[\mathbf{y}_i \mathbf{y}_i^T\right] & \mathrm{E}\left[\mathbf{y}_i\right] \\ \mathrm{E}\left[\mathbf{y}_i\right]^T & 1 \end{bmatrix} \tag{A.119}$$

$$\mathrm{E}\left[\mathbf{x}_{ij}\right] = \mathrm{E}_{\mathbf{Y}}\left[\overline{\mathbf{x}}_{ij}\right] \tag{A.120}$$

$$= \mathbf{L}_{\mathbf{x}_{ij}}^{-1}\mathbf{U}_j^T \mathrm{E}\left[\mathbf{W}\right]\mathrm{E}\left[\theta_{ij}\right]\left(\boldsymbol{\phi}_j - \mathrm{E}\left[\boldsymbol{\mu}\right] - \mathrm{E}\left[\mathbf{V}\right]\mathrm{E}\left[\mathbf{y}_i\right]\right) \tag{A.121}$$

$$\mathrm{E}_{\mathbf{X}}\left[\mathbf{x}_{ij}\mathbf{x}_{ij}^T\right] = \mathrm{E}_{\mathbf{Y}}\left[\mathbf{L}_{\mathbf{x}_{ij}}^{-1} + \overline{\mathbf{x}}_{ij}\overline{\mathbf{x}}_{ij}^T\right] \tag{A.122}$$

$$= \mathbf{L}_{\mathbf{x}_{ij}}^{-1} + \mathbf{L}_{\mathbf{x}_{ij}}^{-1}\mathbf{U}_j^T \mathrm{E}\left[\theta_{ij}\right]^2 \mathrm{E}\left[\mathbf{W}\right]\mathrm{E}_{\mathbf{Y}}\left[\mathbf{G}_{ij}\right]\mathrm{E}\left[\mathbf{W}\right]\mathbf{U}_j^T\mathbf{L}_{\mathbf{x}_{ij}}^{-1} \tag{A.123}$$

$$\mathrm{E}_{\mathbf{Y}}\left[\mathbf{G}_{ij}\right] = \boldsymbol{\phi}_j\boldsymbol{\phi}_j^T - \mathrm{E}\left[\tilde{\mathbf{V}}\right]\mathrm{E}\left[\tilde{\mathbf{y}}_i\right]\boldsymbol{\phi}_j^T - \boldsymbol{\phi}_j\mathrm{E}\left[\tilde{\mathbf{y}}_i\right]^T\mathrm{E}\left[\tilde{\mathbf{V}}\right]^T + \mathrm{E}\left[\tilde{\mathbf{V}}\right]\mathrm{E}\left[\tilde{\mathbf{y}}_i\tilde{\mathbf{y}}_i^T\right]\mathrm{E}\left[\tilde{\mathbf{V}}\right]^T \tag{A.124}$$

$$\mathrm{E}_{\mathbf{XY}}\left[\mathbf{x}_{ij}\mathbf{y}_i^T\right] = \mathrm{E}_{\mathbf{Y}}\left[\overline{\mathbf{x}}_{ij}\mathbf{y}_i^T\right] \tag{A.125}$$

$$= \mathrm{E}_{\mathbf{Y}}\left[\mathbf{L}_{\mathbf{x}_{ij}}^{-1}\mathbf{U}_j^T\mathbf{W}\mathrm{E}\left[\theta_{ij}\right]\left(\boldsymbol{\phi}_j - \boldsymbol{\mu} - \mathbf{V}\mathbf{y}_i\right)\mathbf{y}_i^T\right] \tag{A.126}$$

$$= \mathbf{L}_{\mathbf{x}_{ij}}^{-1}\mathbf{U}_j^T\mathrm{E}\left[\mathbf{W}\right]\mathrm{E}\left[\theta_{ij}\right]\left[(\boldsymbol{\phi}_j - \boldsymbol{\mu})\overline{\mathbf{y}}_i^T - \mathbf{V}\mathrm{E}\left[\mathbf{y}_i\mathbf{y}_i^T\right]\right] \tag{A.127}$$

$$\mathrm{E}\left[\mathbf{x}_{ij}\tilde{\mathbf{y}}_i^T\right] = \left[\mathrm{E}\left[\mathbf{x}_{ij}\mathbf{y}_i^T\right] \quad \mathrm{E}\left[\mathbf{x}_{ij}\right]\right] \tag{A.128}$$

$$\mathrm{E}\left[\theta_{ij}\right] = r_{ij} \tag{A.129}$$

$$\mathrm{E}\left[\theta_{ij}\theta_{ij}\right]\Theta = r_{ij} \tag{A.130}$$

$$\mathrm{E}\left[\pi_{\theta_i}\right] = \frac{\tau_i}{\sum_{i=1}^{I}\tau_i} \tag{A.131}$$

$$\mathrm{E}_{\pi_\theta}\left[\ln\pi_{\theta_i}\right] = \psi(\tau_i) - \psi\left(\sum_{i=1}^{I}\tau_i\right) \tag{A.132}$$

$$\mathrm{E}\left[\boldsymbol{\varepsilon}_q\right] = \frac{a'_{\boldsymbol{\varepsilon}}}{b'_{\boldsymbol{\varepsilon}_q}} \tag{A.133}$$

$$\overline{\tilde{\mathbf{V}}} = \mathrm{E}\left[\tilde{\mathbf{V}}\right] = \begin{bmatrix} \overline{\tilde{\mathbf{v}}}_1'^T \\ \dots \\ \overline{\tilde{\mathbf{v}}}_d'^T \end{bmatrix} \tag{A.134}$$

$$\overline{\mathbf{W}} = \mathrm{E}\left[\mathbf{W}\right] = N'\mathbf{K}^{-1} \tag{A.135}$$

$$\mathrm{E}\left[\mathbf{B}_j\right] \approx \mathrm{E}\left[\left(\mathbf{U}_j\mathbf{U}_j^T + \mathrm{E}\left[\mathbf{W}\right]^{-1}\right)^{-1}\right] \tag{A.136}$$

$$\mathrm{E}\left[\ln(|\mathbf{B}_j|)\right] = \mathrm{E}\left[|\ln(\mathbf{W})|\right] - \mathrm{E}\left[\ln|\mathbf{U}_j\mathbf{U}_j^T\mathbf{W} + \mathbf{I}|\right] \tag{A.137}$$

$$\approx \mathrm{E}\left[|\ln(\mathbf{W})|\right] - \ln|\mathbf{U}_j\mathbf{U}_j^T\mathrm{E}\left[\mathbf{W}\right] + \mathbf{I}| \tag{A.138}$$

$$\mathrm{E}\left[\mathbf{v}_q^T\mathbf{v}_q\right] = \sum_{r=1}^{d}\mathrm{E}\left[\mathbf{v}_{rq}'^T\mathbf{v}_{rq}\right] \tag{A.139}$$

$$= \sum_{r=1}^{d}\mathbf{L}_{\tilde{\mathbf{V}}_{rq}} + \overline{\tilde{\mathbf{v}}}_{rq}'^2 \tag{A.140}$$

$$\Sigma_{\tilde{\mathbf{V}}_r} = \begin{bmatrix} \Sigma_{\mathbf{V}_r} & \Sigma_{\mathbf{V}\boldsymbol{\mu}_r} \\ \Sigma^T_{\mathbf{V}\boldsymbol{\mu}_r} & \Sigma_{\boldsymbol{\mu}_r} \end{bmatrix} = \mathbf{L}^{-1}_{\tilde{\mathbf{V}}_r} \tag{A.141}$$

$$\mathrm{E}\left[\mathbf{V}^T\mathbf{B}_j\mathbf{V}\right] = \sum_{r=1}^{d} \nu_{rr}\Sigma_{\mathbf{V}_r} + \mathrm{E}\left[\mathbf{V}\right]^T \mathrm{E}\left[\mathbf{B}_j\right] \mathrm{E}\left[\mathbf{V}\right] \tag{A.142}$$

$$\mathrm{E}\left[\mathbf{V}^T\mathbf{B}_j\boldsymbol{\mu}\right] = \sum_{r=1}^{d} \nu_{rr}\Sigma_{\mathbf{V}\boldsymbol{\mu}_r} + \mathrm{E}\left[\mathbf{V}\right]^T \mathrm{E}\left[\mathbf{B}_j\right] \mathrm{E}\left[\boldsymbol{\mu}\right] \tag{A.143}$$

$$\mathrm{E}\left[\tilde{\mathbf{V}}^T\mathbf{B}_j\tilde{\mathbf{V}}\right] = \sum_{r=1}^{d} \nu_{rr}\Sigma_{\tilde{\mathbf{V}}_r} + \mathrm{E}\left[\tilde{\mathbf{V}}\right]^T \mathrm{E}\left[\mathbf{B}_j\right] \mathrm{E}\left[\tilde{\mathbf{V}}\right] \tag{A.144}$$

$$\begin{aligned} \mathrm{E}\left[(\boldsymbol{\phi}_j - \tilde{\mathbf{V}}\tilde{\mathbf{y}})^T\mathbf{B}_j(\boldsymbol{\phi}_j - \tilde{\mathbf{V}}\tilde{\mathbf{y}})\right] =& \boldsymbol{\phi}_j^T \mathrm{E}\left[\mathbf{B}_j\right] \boldsymbol{\phi}_j - 2\boldsymbol{\phi}_j \mathrm{E}\left[\mathbf{B}_j\right] \mathrm{E}\left[\tilde{\mathbf{V}}\right]^T \mathrm{E}\left[\tilde{\mathbf{y}}\right] \\ &+ \mathrm{tr}\left(\mathrm{E}\left[\tilde{\mathbf{V}}^T\mathbf{B}_j\tilde{\mathbf{V}}\right] \mathrm{E}\left[\tilde{\mathbf{y}}\tilde{\mathbf{y}}^T\right]\right) \end{aligned} \tag{A.145}$$

$$\mathrm{E}\left[\tilde{\mathbf{V}}\mathbf{R}_{\tilde{\mathbf{y}}}\tilde{\mathbf{V}}^T\right] = \mathrm{E}\left[\tilde{\mathbf{V}}\right] \mathbf{R}_{\tilde{\mathbf{y}}} \mathrm{E}\left[\tilde{\mathbf{V}}\right]^T + \mathrm{diag}(\kappa) \tag{A.146}$$

where

$$\kappa = \begin{bmatrix} \kappa_1 \\ \dots \\ \kappa_d \end{bmatrix} \tag{A.147}$$

$$\kappa_i = \sum_{r=1}^{n_y}\sum_{s=1}^{ny} \left(\mathbf{R}_{\tilde{\mathbf{y}}} \circ \mathbf{L}^{-1}_{\tilde{\mathbf{V}}_i}\right)_{rs} \tag{A.148}$$

denoting $\circ$ the Hadamard product.

If deterministic annealing k is considered, the previous factors are modified in the following way

$$q^*\left(\mathbf{Y}, \mathbf{X}\right) = \prod_{i=1}^{I} \mathcal{N}\left(\mathbf{y}_i | \overline{\mathbf{y}}_i, \frac{1}{k}\mathbf{L}^{-1}_{\mathbf{y}_i}\right) \prod_{j=1}^{N} \mathcal{N}\left(\mathbf{x}_{ij} | \overline{\mathbf{x}}_{ij}, \frac{1}{k}\mathbf{L}^{-1}_{\mathbf{x}_{ij}}\right) \tag{A.149}$$

$$q^*\left(\Theta\right) = \prod_{i=1}^{I}\prod_{j=1}^{N} r_{ij}^{\theta_{ij}}; \quad r_{ij} = \frac{\varrho_{ij}^k}{\sum_{i=1}^{I} \varrho_{ij}^k} \tag{A.150}$$

$$q^*\left(\pi_\theta\right) = C(\tau) \prod_{i=1}^{I} \pi_{\theta_i}^{\tau_i - 1}; \quad \tau_i = k\left(\mathrm{E}_\Theta\left[N_i\right] + \tau_0 - 1\right) + 1 \tag{A.151}$$

$$q^*\left(\tilde{\mathbf{v}}'_r\right) = \mathcal{N}\left(\tilde{\mathbf{v}}'_r | \overline{\tilde{\mathbf{v}}}'_r, \frac{1}{k}\mathbf{L}^{-1}_{\tilde{\mathbf{V}}_r}\right) \tag{A.152}$$

$$q^*\left(\mathbf{W}\right) = \mathcal{W}\left(\mathbf{W} | \frac{1}{k}\mathbf{K}^{-1}, k(N-d-1)+d+1\right); \quad \text{if } \; k(N-d-1)+d+1 > 0 \tag{A.153}$$

$$q^*(\boldsymbol{\varepsilon}) = \prod_{q=1}^{n_y} \mathcal{G}\left(\varepsilon_q | a'_{\boldsymbol{\varepsilon}}, b'_{\varepsilon_q}\right) \tag{A.154}$$

$$a'_{\boldsymbol{\varepsilon}} = k\left(a_{\boldsymbol{\varepsilon}} + \frac{d}{2} - 1\right) + 1 \tag{A.155}$$

$$b'_{\varepsilon_q} = k\left(b_{\boldsymbol{\varepsilon}} + \frac{1}{2}\mathrm{E}\left[\mathbf{v}_q^T \mathbf{v}_q\right]\right) \tag{A.156}$$

A.4.10 Variational Lower Bound

The lower bound for this model is:

$$\begin{aligned}\mathcal{L} =& \mathrm{E}_{\mathbf{Y},\Theta,\mathbf{X},\boldsymbol{\mu},\mathbf{V},\mathbf{W}}\left[\ln P\left(\boldsymbol{\Phi}|\mathbf{Y},\Theta,\mathbf{X},\boldsymbol{\mu},\mathbf{V},\mathbf{W}\right)\right] + \mathrm{E}\left[\ln P\left(\mathbf{Y}\right)\right] + \mathrm{E}\left[\ln P\left(\mathbf{X}\right)\right] \\ &+ \mathrm{E}\left[\ln P\left(\Theta|\pi_\theta\right)\right] + \mathrm{E}\left[\ln P\left(\pi_\theta\right)\right] + \mathrm{E}\left[\ln P\left(\mathbf{V}|\boldsymbol{\varepsilon}\right)\right] + \mathrm{E}\left[\ln P\left(\boldsymbol{\mu}\right)\right] \\ &+ \mathrm{E}\left[\ln P\left(\mathbf{W}\right)\right] + \mathrm{E}\left[\ln P\left(\boldsymbol{\varepsilon}\right)\right] - \mathrm{E}\left[\ln q\left(\mathbf{Y},\mathbf{X}\right)\right] - \mathrm{E}\left[\ln q\left(\Theta\right)\right] \\ &- \mathrm{E}\left[\ln q\left(\pi_\theta\right)\right] - \mathrm{E}\left[\ln q\left(\tilde{\mathbf{V}}\right)\right] - \mathrm{E}\left[\ln q\left(\mathbf{W}\right)\right] - \mathrm{E}\left[\ln q\left(\boldsymbol{\varepsilon}\right)\right]\end{aligned} \tag{A.157}$$

Then, the terms $\mathrm{E}\left[\ln P\left(\cdot\right)\right]$ are defined as:

$$\begin{aligned}\mathrm{E}_{\mathbf{Y},\Theta,\mathbf{X},\mathbf{V},\mathbf{W}}\left[\ln P\left(\boldsymbol{\Phi}|\mathbf{Y},\Theta,\mathbf{X},\boldsymbol{\mu},\mathbf{V},\mathbf{W}\right)\right] =& \frac{\mathrm{E}_\Theta\left[N\right]}{2}\mathrm{E}\left[\ln|\mathbf{W}|\right] - \frac{\mathrm{E}_\Theta\left[N\right]d}{2}\ln 2\pi \\ &- \frac{1}{2}\mathrm{tr}\left(\mathrm{E}\left[\mathbf{W}\right]\left(\mathrm{E}_\Theta\left[\mathbf{S}\right] - 2\mathrm{E}\left[\mathbf{V}\right]\left(\mathbf{C}_{\tilde{\mathbf{y}}} - \mathbf{C}_{\mathbf{x}\tilde{\mathbf{y}}}\right)\right.\right. \\ &\left.\left. -2\mathbf{C}_{\mathbf{x}} + \mathbf{R}_{\mathbf{x}} + \mathrm{E}\left[\tilde{\mathbf{V}}\mathbf{R}_{\tilde{\mathbf{y}}}\tilde{\mathbf{V}}^T\right]\right)\right)\end{aligned} \tag{A.158}$$

Other terms are:

$$\mathrm{E}\left[\ln P\left(\mathbf{Y}\right)\right] = \frac{In_y}{2}\ln(2\pi) - \frac{1}{2}\mathrm{tr}\left(\sum_{i=1}^{I}\mathrm{E}\left[\mathbf{y}_i\mathbf{y}_i^T\right]\right) \tag{A.159}$$

$$\mathrm{E}\left[\ln P\left(\Theta|\pi_\theta\right)\right] = \sum_{i=1}^{I}\sum_{j=1}^{N} r_{ij}\mathrm{E}\left[\ln \pi_{\theta_i}\right] \tag{A.160}$$

$$\mathrm{E}\left[\ln P\left(\pi_\theta\right)\right] = \ln C(\tau_0) + (\tau_0 - 1)\sum_{i=1}^{I}\mathrm{E}\left[\ln \pi_{\theta_i}\right] \tag{A.161}$$

$$\mathrm{E}\left[\ln P\left(\mathbf{X}\right)\right] = \frac{INn_x}{2}\ln(2\pi) - \frac{1}{2}\mathrm{tr}\left(\sum_{i=1}^{I}\sum_{j=1}^{N}\mathrm{E}\left[\mathbf{x}_{ij}\mathbf{x}_{ij}^T\right]\right) \tag{A.162}$$

$$\mathrm{E}\left[\ln P\left(\mathbf{V}|\boldsymbol{\varepsilon}\right)\right] = -\frac{n_y d}{2}\ln(2\pi) + \frac{d}{2}\sum_{q=1}^{n_y}\mathrm{E}\left[\ln \varepsilon_q\right] - \frac{1}{2}\sum_{q=1}^{n_y}\mathrm{E}\left[\varepsilon_q\right]\mathrm{E}\left[\mathbf{v}_q^T\mathbf{v}_q\right] \tag{A.163}$$

$$\mathrm{E}\left[\ln P\left(\boldsymbol{\mu}\right)\right] = -\frac{d}{2}\ln(2\pi) + \frac{1}{2}\sum_{r=1}^{d}\ln\beta_r - \frac{1}{2}\sum_{r=1}^{d}\beta_r\left(\mathrm{E}\left[\boldsymbol{\mu}_r^2\right] - 2\boldsymbol{\mu}_{0_r}\mathrm{E}\left[\boldsymbol{\mu}_r\right] + \boldsymbol{\mu}_{0_r}^2\right) \tag{A.164}$$

$$\mathrm{E}\left[\ln P\left(\mathbf{W}\right)\right] = \ln B\left(\boldsymbol{\Psi}_0, \xi_0\right) + \frac{\xi - d - 1}{2}\ln\mathrm{E}\left[\mathbf{W}\right] - \frac{\xi}{2}\mathrm{tr}\left(\boldsymbol{\Psi}_0^{-1}\boldsymbol{\Psi}\right) \tag{A.165}$$

$$\mathrm{E}\left[\ln P\left(\boldsymbol{\varepsilon}\right)\right] = n_y(a_{\boldsymbol{\varepsilon}}\ln b_{\boldsymbol{\varepsilon}} - \ln\Gamma(a_{\boldsymbol{\varepsilon}})) + \sum_{q=1}^{n_y}(a_{\boldsymbol{\varepsilon}} - 1)\mathrm{E}\left[\ln\boldsymbol{\varepsilon}_q\right] - b_{\boldsymbol{\varepsilon}}\mathrm{E}\left[\boldsymbol{\varepsilon}_q\right] \tag{A.166}$$

Regarding the terms $\mathrm{E}\left[\ln q\left(\cdot\right)\right]$, we can define them as:

$$\begin{aligned}\mathrm{E}\left[\ln q\left(\mathbf{Y}, \mathbf{X}\right)\right] =& -\frac{In_y}{2}(\ln(2\pi) + 1) + \sum_{i=1}^{I}\ln|\mathbf{L}_{\mathbf{y}_i}| \\ & - \frac{INn_x}{2}(\ln(2\pi) + 1) + \sum_{i=1}^{I}\sum_{j=1}^{N}\ln|\mathbf{L}_{\mathbf{x}_{ij}}|\end{aligned} \tag{A.167}$$

$$\mathrm{E}\left[\ln q\left(\Theta\right)\right] = \sum_{i=1}^{I}\sum_{j=1}^{N} r_{ij}\ln r_{ij} \tag{A.168}$$

$$\mathrm{E}\left[\ln q\left(\pi_\theta\right)\right] = \ln C(\tau) + \sum_{i=1}^{I}(\tau_i - 1)\mathrm{E}\left[\ln\pi_{\theta_i}\right] \tag{A.169}$$

$$\mathrm{E}\left[\ln q\left(\tilde{\mathbf{V}}\right)\right] = -\frac{d(n_y + 1)}{2}(\ln(2\pi) + 1) + \frac{1}{2}\sum_{r=1}^{d}\ln\left|\mathbf{L}_{\tilde{\mathbf{V}}_r}\right| \tag{A.170}$$

$$\mathrm{E}\left[\ln q\left(\mathbf{W}\right)\right] = \ln B(\boldsymbol{\Psi}, \xi) + \frac{\xi - d - 1}{2}\ln\mathrm{E}\left[\mathbf{W}\right] - \frac{\xi d}{2} \tag{A.171}$$

$$\mathrm{E}\left[\ln q\left(\boldsymbol{\varepsilon}\right)\right] = \sum_{q=1}^{n_y}(a'_{\boldsymbol{\varepsilon}} - 1)\psi(a_{\boldsymbol{\varepsilon}'}) + \ln b'_{\boldsymbol{\varepsilon}_q} - a'_{\boldsymbol{\varepsilon}} - \ln\Gamma(a_{\boldsymbol{\varepsilon}}) \tag{A.172}$$

A.5 Hyperparameter optimization

For the hyperparameter optimization, we will do it by maximization of the lower bound. so then First, we derive $\mathcal{L}$ with respect to τ_0:

$$\frac{\partial\mathcal{L}}{\partial\tau_0} = \frac{\partial\ln C(\tau_0) + (\tau_0 - 1)\sum_{i=1}^{I}\mathrm{E}_{\pi_\theta}\left[\ln\pi_\theta\right]}{\partial\tau_0} \tag{A.173}$$

$$= \frac{\partial\ln\Gamma(I\tau_0) - I\ln\Gamma(\tau_0) + (\tau_0 - 1)\sum_{i=1}^{I}\mathrm{E}_{\pi_\theta}\left[\ln\pi_\theta\right]}{\partial\tau_0} \tag{A.174}$$

$$= I(\psi(I\tau_0) - \psi(\tau_0)) + \sum_{i=1}^{I}\mathrm{E}_{\pi_\theta}\left[\ln\pi_\theta\right] \tag{A.175}$$

$$\frac{\partial \mathcal{L}}{\partial \tau_0} = 0 => \tag{A.176}$$

$$(\psi(I\tau_0) - \psi(\tau_0)) + \frac{1}{I}\sum_{i=1}^{I} \mathrm{E}_{\pi_\theta}\left[\ln \pi_\theta\right] = f(\tau_0) \tag{A.177}$$

The previous equation is hard to deal with, so in order to achieve estimate its update we will apply the Newton-Rhapson iterative procedure. First we define a new variable $\tau_0 = exp(\tilde{\tau}_0)$

$$\tilde{\tau}_{0_{new}} = \frac{f(\tilde{\tau}_0)}{f'(\tilde{\tau}_0)} = \tilde{\tau}_0 - \frac{\psi(I\tau_0) - \psi(\tau_0) + \frac{1}{I}\sum_{i=1}^{I} \mathrm{E}_{\pi_\theta}\left[\ln \pi_\theta\right]}{\tau_0(\psi'(I\tau_0) - \psi'(\tau_0))} \tag{A.178}$$

Taking exponentials in both sides

$$\tau_{0_{new}} = \tau_0 exp\left(-\frac{\psi(I\tau_0) - \psi(\tau_0) + \frac{1}{I}\sum_{i=1}^{I} \mathrm{E}_{\pi_\theta}\left[\ln \pi_\theta\right]}{\tau_0(\psi'(I\tau_0) - \psi'(\tau_0))}\right) \tag{A.179}$$

We derive also for $a_{\boldsymbol{\varepsilon}}$

$$\frac{\partial \mathcal{L}}{\partial a_{\boldsymbol{\varepsilon}}} = n_y(\ln(b_{\boldsymbol{\varepsilon}}) - \psi(a_{\boldsymbol{\varepsilon}}) + \sum_{q=1}^{n_y} \mathrm{E}\left[\ln(\boldsymbol{\varepsilon}_q)\right]) \tag{A.180}$$

$$\frac{\partial \mathcal{L}}{\partial a_{\boldsymbol{\varepsilon}}} = 0 => \tag{A.181}$$

$$\psi(a_{\boldsymbol{\varepsilon}}) = \ln(b_{\boldsymbol{\varepsilon}}) + \frac{1}{n_y}\sum_{q=1}^{n_y} \mathrm{E}\left[\ln(\boldsymbol{\varepsilon}_q)\right] \tag{A.182}$$

Deriving for $b_{\boldsymbol{\varepsilon}}$

$$\frac{\partial \mathcal{L}}{\partial b_{\boldsymbol{\varepsilon}}} = \frac{n_y a_{\boldsymbol{\varepsilon}}}{b_{\boldsymbol{\varepsilon}}} - \sum_{q=1}^{n_y} \mathrm{E}\left[\ln(\boldsymbol{\varepsilon}_q)\right] \tag{A.183}$$

$$\frac{\partial \mathcal{L}}{\partial b_{\boldsymbol{\varepsilon}}} = 0 => \tag{A.184}$$

$$b_{\boldsymbol{\varepsilon}} = \left(\frac{1}{n_y a_{\boldsymbol{\varepsilon}}}\sum_{q=1}^{n_y} \mathrm{E}\left[\ln(\boldsymbol{\varepsilon}_q)\right]\right)^{-1} \tag{A.185}$$

In order to solve both equations, we express

$$\psi(a) = \ln(b) + c \tag{A.186}$$

$$b = \frac{a}{d} \tag{A.187}$$

where

$$c = \frac{1}{n_y}\sum_{q=1}^{n_y} \mathrm{E}\left[\ln(\boldsymbol{\varepsilon}_q)\right] \tag{A.188}$$

$$d = \frac{1}{n_y} \sum_{q=1}^{n_y} \mathrm{E}\left[\boldsymbol{\varepsilon}_q\right] \tag{A.189}$$

Then

$$f(a) = \psi(a) - \ln(a) + \ln(d) - c = 0 \tag{A.190}$$

By solving a by means of Newton Rhapson iterations

$$a_{new} = a - \frac{f(a)}{f'(a)} \tag{A.191}$$

$$= a\left(1 - \frac{\psi(a) - \ln(a) + \ln(d) - c}{a\psi'(a) + 1}\right) \tag{A.192}$$

In order to reassure a positive value for a we can solve for $\tilde{a}$ as $a = \exp(\tilde{a})$

$$\tilde{a}_{new} = \tilde{a} - \frac{f(\tilde{a})}{f'(\tilde{a})} \tag{A.193}$$

$$= \tilde{a}\left(1 - \frac{\psi(a) - \ln(a) + \ln(d) - c}{a\psi'(a) + 1}\right) \tag{A.194}$$

Taking exponentials

$$a = a\exp\left(1 - \frac{\psi(a) - \ln(a) + \ln(d) - c}{a\psi'(a) + 1}\right) \tag{A.195}$$

Now deriving for $\boldsymbol{\mu}_0$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}_0} = 0 => \tag{A.196}$$

$$\boldsymbol{\mu}_0 = \mathrm{E}\left[\boldsymbol{\mu}\right] \tag{A.197}$$

Finally, the optimization of β.

$$\frac{\partial \mathcal{L}}{\partial \beta} = 0 => \tag{A.198}$$

$$\beta_r^{-1} = \frac{1}{d} \sum_{r=1}^{d} \Sigma_{\boldsymbol{\mu}_r} + \mathrm{E}\left[\boldsymbol{\mu}_r\right]^2 - 2\boldsymbol{\mu}_{0_r} \mathrm{E}\left[\boldsymbol{\mu}_r\right] + \boldsymbol{\mu}_{0_r}^2 \tag{A.199}$$

Bibliography

[Ajili et al., 2016] Ajili, M., Bonastre, J.-F., Waad, B. K., Solange, R., and Juliette, K. (2016). Phonetic content impact on Forensic Voice Comparison. *IEEE Workshop on Spoken Language Technology (SLT)*, pages 210–217.

[Ajmera and Wooters, 2003] Ajmera, J. and Wooters, C. (2003). A robust speaker clustering algorithm. In *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 413–416.

[Alam et al., 2015] Alam, M. J., Kenny, P., and Stafylakis, T. (2015). Combining amplitude and phase-based features for speaker verification with short duration utterances. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 249–253.

[Alam et al., 2011] Alam, M. J., Ouellet, P., Kenny, P., and O'Shaughnessy, D. (2011). Comparative evaluation of feature normalization techniques for speaker verification. *Advances in Nonlinear Speech Processing. NOLISP 2011. Lecture Notes in Computer Science*, 7015(2011):246–253.

[Anguera et al., 2012] Anguera, X., Bozonnet, S., Evans, N., Fredouille, C., Friedland, G., and Vinyals, O. (2012). Speaker Diarization: A Review of Recent Research. *IEEE Transactions On Audio Speech And Language Processing*, 20(2):356–370.

[Attias, 1999] Attias, H. (1999). Inferring Parameters and Structure of Latent Variable Models by Variational Bayes. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, volume 1, pages 21–30.

[Beek et al., 1977] Beek, B., Neuberg, E. P., and Hodge, D. c. (1977). An Assessment of the Technology of Automatic Speech Recognition for Military Applications. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 25(4):310–322.

[Bell et al., 2015] Bell, P., Gales, M. J. F., Hain, T., Kilgour, J., Lanchantin, P., Liu, X., McParland, A., Renals, S., Saz, O., Wester, M., and Woodland, P. C. (2015). The MGB Challenge: Evaluating Multi-Genre Broadcast Media Recognition. In *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 687–693.

[Ben et al., 2004] Ben, M., Betser, M., Bimbot, F., and Gravier, G. (2004). Speaker diarization using bottom-up clustering based on a parameter-derived distance between adapted GMMs. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 2329–2332.

[Bimbot et al., 2004] Bimbot, F., Bonastre, J.-F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-García, J., Petrovska-Delacrétaz, D., and Reynolds, D. A. (2004). A Tutorial on Text-Independent Speaker Verification. *EURASIP Journal on Advances in Signal Processing*, 2004(4):430–451.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., New York, NY, USA.

[Blei and Frazier, 2011] Blei, D. M. and Frazier, P. I. (2011). Distance Dependent Chinese Restaurant Processes. *Journal of Machine Learning Research (JMLR)*, 12:2461–2488.

[Brummer et al., 2017] Brummer, N., Burget, L., Garcia, P., Plchot, O., Rohdin, J., Garcia-Romero, D., Snyder, D., Stafylakis, T., Swart, A., and Villalba, J. (2017). Meta-embeddings : a probabilistic generalization of embeddings in machine learning. Technical report.

[Brummer and de Villiers, 2010] Brummer, N. and de Villiers, E. (2010). The Speaker Partitioning Problem. In *ODYSSEY The Speaker and Language Recognition Workshop*, pages 194–201.

[Brummer and Du Preez, 2006] Brummer, N. and Du Preez, J. (2006). Application-independent evaluation of speaker detection. *Computer Speech and Language*, 20(2-3 SPEC. ISS.):230–275.

[Brummer et al., 2018] Brummer, N., Silnova, A., Burget, L., and Stafylakis, T. (2018). Gaussian meta-embeddings for efficient scoring of a heavy-tailed PLDA model. *ODYSSEY The Speaker and Language Recognition Workshop*, pages 349–356.

[Calvo et al., 2007] Calvo, J. R., Fernández, R., and Hernández, G. (2007). Application of shifted delta cepstral features in speaker verification. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 1, pages 29–32.

[Campbell et al., 2006a] Campbell, W. M., Sturim, D. E., and Reynolds, D. A. (2006a). Support vector machines using GMM supervectors for speaker verification. *IEEE Signal Processing Letters*, 13(5):308–311.

[Campbell et al., 2006b] Campbell, W. M., Sturim, D. E., Reynolds, D. A., and Solomonoff, A. (2006b). SVM Based Speaker Verification using a GMM Supervector Kernel and NAP Variability Compensation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 97–100, Toulouse, France. IEEE.

[Chen and Gopalakrishnan, 1998] Chen, S. S. and Gopalakrishnan, P. (1998). Speaker, Environment and Channel Change Detection and Clustering Via the Bayesian Information Criterion. *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, 6:127–132.

[Chung et al., 2018] Chung, J. S., Nagrani, A., and Zisserman, A. (2018). Voxceleb2: Deep Speaker Recognition. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2018-Septe(ii):1086–1090.

[Comaniciu and Meer, 2002] Comaniciu, D. and Meer, P. (2002). Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619.

[Cumani et al., 2013a] Cumani, S., Brummer, N., Burget, L., Laface, P., Plchot, O., and Vasilakakis, V. (2013a). Pairwise discriminative speaker verification in the I-vector space. *IEEE Transactions on Audio, Speech and Language Processing*, 21(6):1217–1227.

[Cumani et al., 2013b] Cumani, S., Plchot, O., and Laface, P. (2013b). Probabilistic linear discriminant analysis of i-vector posterior distributions. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7644–7648.

[Davis and Mermelstein, 1980] Davis, S. B. and Mermelstein, P. (1980). Comparison of Parametric Representations for. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):357–366.

[Dehak et al., 2011] Dehak, N., Kenny, P., Dehak, R., Dumouchel, P., and Ouellet, P. (2011). Front-End Factor Analysis For Speaker Verification. *IEEE Transactions on Audio, Speech and Language Processing*, 19(4):788–798.

[Delacourt and Wellekens, 2000] Delacourt, P. and Wellekens, C. J. (2000). DISTBIC: A speaker-based segmentation for audio data indexing. *Speech Communication*, 32(1):111–126.

[Diez et al., 2018] Diez, M., Burget, L., and Matejka, P. (2018). Speaker Diarization based on Bayesian HMM with Eigenvoice Priors. *ODYSSEY The Speaker and Language Recognition Workshop*, pages 147–154.

[Diez et al., 2019] Diez, M., Burget, L., Wang, S., Rohdin, J., and Cernocký, H. (2019). Bayesian HMM based x-vector clustering for Speaker Diarization. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 346–350.

[Doddington, 1971] Doddington, G. R. (1971). A Method or Speaker Verification. *The Journal of the Acoustical Society of America*, 49(1A):139.

[Duda and Hart, 1973] Duda, R. O. and Hart, P. E. (1973). *Pattern Classification and Scene Analysis*.

[Friedland et al., 2009] Friedland, G., Vinyals, O., Huang, Y., and Müller, C. (2009). Prosodic and other long-term features for speaker diarization. *IEEE Transactions on Audio, Speech and Language Processing*, 17(5):985–993.

[Fukunaga and Hostetler, 1975] Fukunaga, K. and Hostetler, L. (1975). The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition. *IEEE Transactions on Information Theory*, 21(1):32–40.

[Furui, 1981] Furui, S. (1981). Cepstral analysis technique for automatic speaker verification. *IEEE Transactions on Acoustics Speech and Signal Processing*, 29(2):254–272.

[Furui, 2004] Furui, S. (2004). Fifty years of progress in speech and speaker recognition. *The Journal of the Acoustical Society of America*, 116(4):2497–2498.

[Garcia-Romero and Espy-Wilson, 2011] Garcia-Romero, D. and Espy-Wilson, C. Y. (2011). Analysis of I-vector Length Normalization in Speaker Recognition Systems. In *Proceedings*

of the Annual Conference of the International Speech Communication Association, INTERSPEECH, pages 249–252.

[Garcia-Romero et al., 2017] Garcia-Romero, D., Snyder, D., Sell, G., Povey, D., and McCree, A. (2017). Speaker Diarization Using Deep Neural Network Embeddings. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4930 – 4934.

[Garofolo et al., 2002] Garofolo, J., Fiscus, J. G., Martin, A., Pallett, D., and Przybocki, M. (2002). NIST rich transcription 2002 evaluation: A preview. *Proceedings of the 3rd International Conference on Language Resources and Evaluation, LREC 2002*, pages 655–659.

[Garofolo et al., 1993] Garofolo, J. S., Lamel, L. F., Fisher, W. M., Fiscus, J. G., Pallett, D. S., and Dahlgren, N. L. (1993). TIMIT Acoustic-Phonetic Continuous Speech Corpus LDC93S1. pages 1–94.

[Gish et al., 1991] Gish, H., Siu, M.-H., and Rohlicek, R. (1991). Segregation of Speakers for Speech Recognition and Speaker Identification. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 873–876.

[Graves et al., 2013] Graves, A., Mohamed, A.-r., and Hinton, G. E. (2013). Speech recognition with deep recurrent neural networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6645–6649, Vancouver, British Columbia, Canada. IEEE.

[Graves and Schmidhuber, 2005] Graves, A. and Schmidhuber, J. (2005). Framewise Phoneme Classification with Bidirectional LSTM and other Neural Network Architectures. *Neural Networks*, 18(5-6):602–610.

[Greenberg et al., 2011] Greenberg, C. S., Martin, A. F., Barr, B. N., and Doddington, G. R. (2011). Report on performance results in the NIST 2010 Speaker Recognition Evaluation. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 261–264.

[Guo et al., 2017] Guo, J., Xu, N., Li, L. J., and Alwan, A. (2017). Attention based CLDNNs for short-duration acoustic scene classification. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2017-Augus:469–473.

[Gupta, 2015] Gupta, V. (2015). Speaker change point detection using deep neural nets. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4420–4424.

[Hamidi Ghalehjegh and Rose, 2015] Hamidi Ghalehjegh, S. and Rose, R. C. (2015). Deep Bottleneck Features For I-Vector Based Text-Independent Speaker Verification. *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 555–560.

[Hasan et al., 2013] Hasan, T., Saeidi, R., Hansen, J. H. L., and Van Leeuwen, D. A. (2013). Duration mismatch compensation for i-vector based speaker recognition systems. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (238803):7663–7667.

[Hautamäki et al., 2013] Hautamäki, V., Cheng, Y. C., Rajan, P., and Lee, C. H. (2013). Minimax i-vector extractor for short duration speaker verification. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 3708–3712.

[Hermansky, 1990] Hermansky, H. (1990). Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustical Society of America*, 87(4):1738–1752.

[Hinton et al., 2012] Hinton, G. E., Deng, L., Yu, D., Dahl, G., Mohamed, A.-r., Jaitly, N., Vanhoucke, V., Nguyen, P., Sainath, T., and Kingsbury, B. (2012). Deep Neural Networks for Acoustic Modeling in Speech Recognition. *IEEE Signal Processing Magazine*, 29(6):82–97.

[Hruz and Zajic, 2017] Hruz, M. and Zajic, Z. (2017). Convolutional Neural Network for Speaker Change Detection in Telephone Speaker Diarization System. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4945–4949.

[Huang and Hansen, 2006] Huang, R. and Hansen, J. H. L. (2006). Advances in unsupervised audio segmentation for the broadcast news and NGSW corpora. *IEEE Transactions on Audio, Speech and Language Processing*, 14(3):907–919.

[Ioffe and Szegedy, 2015] Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *32nd International Conference on Machine Learning, ICML 2015*, volume 1, pages 448–456.

[Jelinek, 1976] Jelinek, F. (1976). Continuous Speech Recognition by Statistical Methods. *Proceedings of the IEEE*, 64(4):532–556.

[Jelinek and Anderson, 1971] Jelinek, F. and Anderson, J. (1971). Instrumentable Tree Encoding of Information Sources. *IEEE Transactions on Information Theory*, (January):118–119.

[Jin et al., 1997] Jin, H., Kubala, F., and Schwartz, R. (1997). Automatic Speaker Clustering. In *Proc. DARPA Speech Recognition Workshop*, pages 108–111.

[Juang, 1990] Juang, B.-H. (1990). Speaker Recognition Based on Source Coding Approaches. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 613–616.

[Julier and Uhlmann, 2004] Julier, S. J. and Uhlmann, J. K. (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(12):401–422.

[Kanagasundaram et al., 2013] Kanagasundaram, A., Dean, D., Gonzalez-Dominguez, J., Sridharan, S., Ramos, D., and Gonzalez-Rodriguez, J. (2013). Improving short utterance based I-vector speaker recognition using source and utterance-duration normalization techniques. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, (August):2465–2469.

[Kanagasundaram et al., 2011] Kanagasundaram, A., Dean, D., Sridharan, S., and Fookes, C. (2011). Domain adaptation based Speaker Recognition on Short Utterances. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, (August):2341–2344.

[Kanagasundaram et al., 2014] Kanagasundaram, A., Dean, D., Sridharan, S., Gonzalez-Dominguez, J., Gonzalez-Rodriguez, J., and Ramos, D. (2014). Improving short utterance i-vector speaker verification using utterance variance modelling and compensation techniques. *Speech Communication*, 59:69–82.

[Karanasou et al., 2016] Karanasou, P., Gales, M. J. F., Lanchantin, P., Liu, X., Qian, Y., Wang, L., Woodland, P. C., and Zhang, C. (2016). Speaker diarisation and longitudinal linking in multi-genre broadcast data. *IEEE Workshop on Automatic Speech Recognition and Understanding, (ASRU)*, pages 660–666.

[Kenny, 2005] Kenny, P. (2005). Joint Factor Analysis of Speaker and Session Variability: Theory and Algorithms. *CRIM, Montreal,(Report) CRIM-06/08-13*, pages 1–17.

[Kenny, 2010] Kenny, P. (2010). Bayesian Speaker Verification with Heavy-Tailed Priors. *ODYSSEY The Speaker and Language Recognition Workshop*.

[Kenny et al., 2010] Kenny, P., Reynolds, D. A., and Castaldo, F. (2010). Diarization of telephone conversations using factor analysis. *IEEE Journal on Selected Topics in Signal Processing*, 4(6):1059–1070.

[Kenny et al., 2013] Kenny, P., Stafylakis, T., Ouellet, P., Alam, M. J., and Dumouchel, P. (2013). PLDA for Speaker Verification with Utterances of Arbitrary Duration. *Journal of Chemical Information and Modeling*, 53(9):1689–1699.

[Kinnunen and Li, 2010] Kinnunen, T. and Li, H. (2010). An overview of text-independent speaker recognition: From features to supervectors. *Speech Communication*, 52(1):12–40.

[Kullback and Leibler, 1951] Kullback, S. and Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics*, 1(22):79–86.

[Larcher et al., 2012] Larcher, A., Lee, K. A., Ma, B., and Li, H. (2012). The RSR2015: Database for Text-Dependent Speaker Verification using Multiple Pass-Phrases. In *Proceedings of the 13th Annual Conference of the International Speech Communication Association, Interspeech 2012*, Portland, Oregon, USA. ISCA.

[Le Lan et al., 2016] Le Lan, G., Meignier, S., Charlet, D., and Larcher, A. (2016). First investigations on self trained speaker diarization. In *ODYSSEY The Speaker and Language Recognition Workshop*, pages 152–157.

[Lei et al., 2014] Lei, Y., Scheffer, N., Ferrer, L., and McLaren, M. (2014). A Novel Scheme for Speaker Recognition Using a Phonetically-aware Deep Neural Network. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1714–1718.

[Li and Hughes, 1974] Li, K. P. and Hughes, G. W. (1974). Talker differences as they appear in correlation matrices of continuous speech spectra. *The Journal of the Acoustical Society of America*, 55(4):833–837.

[Li et al., 2009] Li, R., Schultz, T., and Jin, Q. (2009). Improving speaker segmentation via speaker identification and text segmentation. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 904–907.

[Li et al., 2015] Li, Z. Y., Zhang, W. Q., and Liu, J. (2015). Multi-resolution time frequency feature and complementary combination for short utterance speaker recognition. *Multimedia Tools and Applications*, 74(3):937–953.

[Lleida et al., 2019] Lleida, E., Ortega, A., Miguel, A., Bazán, V., Pérez, C., Gómez, M., and de Prada, A. (2019). Albayzin 2018 evaluation: The IberSpeech-RTVE challenge on speech technologies for Spanish broadcast media. *Applied Sciences*, 9(24):1–22.

[Lloyd, 1982] Lloyd, S. P. (1982). Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137.

[Lozano-diez et al., 2016] Lozano-diez, A., Silnova, A., Matejka, P., Glembek, O., Plchot, O., Pesan, J., Burget, L., and Gonzalez-Rodriguez, J. (2016). Analysis and Optimization of Bottleneck Features for Speaker Recognition. *ODYSSEY The Speaker and Language Recognition Workshop*, pages 352–357.

[Mandasari et al., 2011] Mandasari, M. I., Mclaren, M., and Van Leeuwen, D. A. (2011). Evaluation of i-vector Speaker Recognition Systems for Forensic Applcation. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, (2):21–24.

[Mandasari et al., 2013] Mandasari, M. I., Saeidi, R., McLaren, M., and Van Leeuwen, D. A. (2013). Quality measure functions for calibration of speaker recognition systems in various duration conditions. *IEEE Transactions on Audio, Speech and Language Processing*, 21(11):2425–2438.

[Martin and Greenberg, 2009] Martin, A. F. and Greenberg, C. S. (2009). NIST 2008 speaker recognition evaluation: Performance across telephone and room microphone channels. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 2579–2582.

[Miasato Filho et al., 2018] Miasato Filho, V. A., Silva, D. A., and Cuozzo, L. G. D. (2018). Joint discriminative embedding learning, speech activity and overlap detection for the DIHARD speaker diarization challenge. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2018-Septe(September):2818–2822.

[Mikolov et al., 2013] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, pages 1–12.

[Morgan et al., 1991] Morgan, N., Wooters, C., and Hermansky, H. (1991). Experiments with temporal resolution for continuous speech recognition with multi-labyer perceptrons. *Neural Networks for Signal Processing Proceedings1*, pages 405–410.

[Nagrani et al., 2017] Nagrani, A., Chung, J. S., and Zisserman, A. (2017). VoxCeleb: A large-scale speaker identification dataset. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2017-Augus:2616–2620.

[Ng. et al., 2002] Ng., A. Y., Jordan, M. I., and Weiss, Y. (2002). On Spectral Clustering: Analysis and an algorithm. *Advances in neural information processing systems*, pages 849–856.

[Nidadavolu et al., 2019] Nidadavolu, P. S., Villalba, J., and Dehak, N. (2019). Cycle-GANs for Domain Adaptation of Acoustic Features for Speaker Recognition. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6206–6210.

[Ning et al., 2006] Ning, H., Liu, M., Tang, H., and Huang, T. (2006). A spectral clustering approach to speaker diarization. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 5:2178–2181.

[NIST, 2008] NIST (2008). The NIST Year 2008 Speaker Recognition Evaluation Plan.

[NIST, 2010] NIST (2010). The NIST Year 2010 speaker recognition evaluation.

[Ortega et al., 2018] Ortega, A., Viñals, I., Miguel, A., Lleida, E., Bazán, V., Perez, C., Gómez, M., and De Prada, A. (2018). Albayzin Evaluation: IberSPEECH-RTVE 2018 Speaker Diarization Challenge. Technical report.

[Otterson and Ostendorf, 2007] Otterson, S. and Ostendorf, M. (2007). Efficient use of overlap information in speaker diarization. In *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 683–686.

[Panayotov et al., 2015] Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). LIBRISPEECH: An ASR Corpus Based on Public Domain Audio Books. In *Proceedings of the*

IEEE International Conference on Acoustics, Speech and Signal Processing, pages 5206–5210.

[Pelecanos and Sridharan, 2001] Pelecanos, J. and Sridharan, S. (2001). Feature Warping for Robust Speaker Verification. *ODYSSEY The Speaker and Language Recognition Workshop*, pages 213–218.

[Poddar et al., 2017] Poddar, A., Sahidullah, M., and Saha, G. (2017). Speaker verification with short utterances: a review of challenges, trends and opportunities. *IET Biometrics*, 7(2):91–101.

[Povey et al., 2011] Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., and Vesely, K. (2011). The Kaldi Speech Recognition Toolkit. *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*.

[Prince and Elder, 2007] Prince, S. J. D. and Elder, J. H. (2007). Probabilistic Linear Discriminant Analysis for Inferences About Identity. In *Proceedings of the IEEE International Conference on Computer Vision*.

[Pruzansky, 1963] Pruzansky, S. (1963). Pattern Matching Procedure for Automatic Talker Recognition. *The Journal of the Acoustical Society of America*, 35(3):354–358.

[Przybocki and Martin, 2004] Przybocki, M. and Martin, A. (2004). NIST Speaker Recognition Evaluation Chronicles. *ODYSSEY The Speaker and Language Recognition Workshop*, pages 12–22.

[Rabiner, 1989] Rabiner, L. R. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2):257–286.

[Ramirez et al., 2004] Ramirez, J., Segura, J. C., Benitez, C., Torre, A. D. L., and Rubio, A. (2004). Voice activity detection with noise reduction and long-term spectral divergence estimation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, pages 1093–1096, Montreal, Quebec, Canada. IEEE.

[Reynolds et al., 2009] Reynolds, D. A., Kenny, P., and Castaldo, F. (2009). A study of New Approaches to Speaker Diarization. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 1047–1050.

[Reynolds et al., 2000] Reynolds, D. A., Quatieri, T. F., and Dunn, R. B. (2000). Speaker verification using adapted Gaussian mixture models. *Digital Signal Processing*, 10(1):19–41.

[Reynolds and Rose, 1995] Reynolds, D. A. and Rose, R. C. (1995). Robust text-independent speaker identification using Gaussian mixture speaker models. *IEEE Transactions On Speech And Audio Processing*, 3(1):72–83.

[Rohdin et al., 2019] Rohdin, J., Stafylakis, T., Silnova, A., Zeilani, H., Burget, L., and Plchot, O. (2019). Speaker verification using End-to-End Adversarial Language Adaptation. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6006–6010.

[Rosenberg and Soong, 1987] Rosenberg, A. E. and Soong, F. K. (1987). Evaluation of a vector quantization talker recognition system in text independent and text dependent modes. *Computer Speech and Language*, 2(3-4):143–157.

[Rousseau et al., 2012] Rousseau, A., Deléglise, P., and Estève, Y. (2012). TED-LIUM: An automatic speech recognition dedicated corpus. *Proceedings of the 8th International Conference on Language Resources and Evaluation, LREC 2012*, pages 125–129.

[Rumelhart et al., 1986] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning Representations by Back-propagationg Errors. *Nature*, 323:533–536.

[Sadjadi et al., 2016] Sadjadi, S. O., Ganapathy, S., and Pelecanos, J. (2016). The IBM 2016 Speaker Recognition System. *ODYSSEY The Speaker and Language Recognition Workshop*, pages 174–180.

[Salmun et al., 2017] Salmun, I., Shapiro, I., Opher, I., and Lapidot, I. (2017). PLDA-Based Mean Shift Speakers' Short Segments Clustering. *Computer Speech and Language*, 45:411–436.

[Sambur, 1972] Sambur, M. R. (1972). *Speaker Recognition and verification using linear prediction analysis*. PhD thesis, M.I.T.

[Sarkar et al., 2012] Sarkar, A. K., Matrouf, D., Bousquet, P.-M., and Bonastre, J.-F. (2012). Study of the Effect of I-vector Modeling on Short and Mismatch Utterance Duration for Speaker Verification. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, (September):2662–2665.

[Schwarz, 1978] Schwarz, G. (1978). Estimating the Dimension of a Model. *The Annals of Statistics*, 6(2):461–464.

[Senoussaoui et al., 2014] Senoussaoui, M., Kenny, P., Stafylakis, T., and Dumouchel, P. (2014). A study of the Cosine Distance-Based Mean Shift for Telephone Speech Diarization. *IEEE Transactions on Audio, Speech and Language Processing*, 22(1):217–227.

[Shriberg et al., 2005] Shriberg, E., Ferrer, L., Kajarekar, S. S., Venkataraman, A., and Stolcke, A. (2005). Modeling prosodic feature sequences for speaker recognition. *Speech Communication*, 46(3-4):455–472.

[Shum et al., 2013] Shum, S. H., Dehak, N., Dehak, R., and Glass, J. R. (2013). Unsupervised methods for speaker diarization: An integrated and iterative approach. *IEEE Transactions on Audio, Speech and Language Processing*, 21(10):2015–2028.

[Siegler et al., 1997] Siegler, M. A., Jain, U., Raj, B., and Stern, R. M. (1997). Automatic Segmentation, Classification and Clustering of Broadcast News Audio. In *Proc. DARPA Speech Recognition Workshop*, pages 97–99.

[Snoek et al., 2012] Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical Bayesian Optimization of Machine Learning Algorithms. *NIPS*, pages 1–9.

[Snyder et al., 2018] Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., and Khudanpur, S. (2018). X-VECTORS : ROBUST DNN EMBEDDINGS FOR SPEAKER RECOGNITION. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5329–5333.

[Snyder et al., 2016] Snyder, D., Ghahremani, P., Povey, D., Garcia-Romero, D., Carmiel, Y., and Khudanpur, S. (2016). Deep Neural Network-based Speaker Embeddings for End-to-end Speaker Verification. In *IEEE Workshop on Spoken Language Technology (SLT)*, pages 165–170.

[Sokal and Michener, 1958] Sokal, R. R. and Michener, C. (1958). *The University of Kansas science bulletin* ., volume 38.

[Soong et al., 1985] Soong, F. K., Rosenberg, A. E., Rabiner, L. R., and Juang, B. H. (1985). Vector Quantization Approach To Speaker Recognition. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 10:387–390.

[Stafylakis et al., 2010] Stafylakis, T., Katsouros, V., and Carayannis, G. (2010). Speaker clustering via the mean shift algorithm. In *ODYSSEY The Speaker and Language Recognition Workshop*, pages 186–193.

[Todisco et al., 2017] Todisco, M., Delgado, H., and Evans, N. (2017). Constant Q cepstral coefficients: A spoofing countermeasure for automatic speaker verification. *Computer Speech and Language*, 45:516–535.

[Tranter and Reynolds, 2006] Tranter, S. E. and Reynolds, D. A. (2006). An overview of automatic speaker diarization systems. *IEEE Transactions on Audio, Speech and Language Processing*, 14(5):1557–1565.

[Valente et al., 2010] Valente, F., Motlicek, P., and Vijayasenan, D. (2010). Variational Bayesian Speaker Diarization of Meeting Recordings. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4954–4957.

[van Leeuwen, 2010] van Leeuwen, D. (2010). Speaker linking in large data sets. In *Proceedings of the Speaker and Language Recognition Odyssey*, pages 202–208.

[Vaquero et al., 2013] Vaquero, C., Ortega, A., Miguel, A., and Lleida, E. (2013). Quality Assessment of Speaker Diarization for Speaker Characterization. *IEEE Transactions on Audio, Speech and Language Processing*, 21(4):816–827.

[Villalba et al., 2019] Villalba, J., Chen, N., Snyder, D., Garcia-Romero, D., McCree, A., Sell, G., Borgstrom, J., Richardson, F., Shon, S., Grondin, F., Dehak, R., Garcia-Perera, L. P., Povey, D., Torres-Carrasquillo, P., Khudanpur, S., and Dehak, N. (2019). State-of-the-art Speaker Recognition for Telephone and Video Speech: the JHU-MIT Submission for NIST SRE18. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 1488–1492.

[Villalba and Lleida, 2014] Villalba, J. and Lleida, E. (2014). Unsupervised Adaptation of PLDA By Using Variational Bayes Methods. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 744–748.

[Villalba et al., 2015] Villalba, J., Ortega, A., Miguel, A., and Lleida, E. (2015). Variational Bayesian PLDA for Speaker Diarization in the MGB Challenge. In *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 667–674.

[Viñals et al., 2018a] Viñals, I., Gimeno, P., Ortega, A., Miguel, A., and Lleida, E. (2018a). Estimation of the Number of Speakers with Variational Bayesian PLDA in the DIHARD Diarization Challenge. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 2803–2807.

[Viñals et al., 2019a] Viñals, I., Gimeno, P., Ortega, A., Miguel, A., and Lleida, E. (2019a). ViVoLAB Speaker Diarization System for the DIHARD 2019 Challenge. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 988–992.

[Viñals et al., 2018b] Viñals, I., Ortega, A., Miguel, A., and Lleida, E. (2018b). Phonetic Variability Influence on Short Utterances in Speaker Verification. *Proceedings of IberSpeech 2018, Advances in Speech and Language Technologies for Iberian Languages*, pages 6–9.

[Viñals et al., 2019b] Viñals, I., Ortega, A., Miguel, A., and Lleida, E. (2019b). An Analysis of the Short Utterance Problem for Speaker Characterization. *Applied Sciences*, 9(18):3697.

[Viñals et al., 2017] Viñals, I., Ortega, A., Villalba, J., Miguel, A., and Lleida, E. (2017). Domain Adaptation of PLDA models in Broadcast Diarization by means of Unsupervised Speaker Clustering. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 2829–2833.

[Viñals et al., 2019c] Viñals, I., Ortega, A., Villalba, J., Miguel, A., and Lleida, E. (2019c). Unsupervised adaptation of PLDA models for broadcast diarization. *Eurasip Journal on Audio, Speech, and Music Processing*, 2019(1).

[Viñals et al., 2019d] Viñals, I., Ribas, D., Mingote, V., Llombart, J., Gimeno, P., Miguel, A., Ortega, A., and Lleida, E. (2019d). Phonetically-aware embeddings, Wide Residual Networks with Time-Delay Neural Networks and Self Attention models for the 2018 NIST Speaker Recognition Evaluation. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 4310–4314.

[Viñals et al., 2016] Viñals, I., Villalba, J., Ortega, A., Miguel, A., and Lleida, E. (2016). Bottleneck based front-end for diarization systems. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10077 LNAI(610986):276–286.

[Viterbi, 1967] Viterbi, A. J. (1967). Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269.

[Vogt et al., 2008] Vogt, R., Baker, B., and Sridharan, S. (2008). Factor analysis subspace estimation for speaker verification with short utterances. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 853–856.

[Vogt et al., 2010] Vogt, R., Sridharan, S., and Mason, M. (2010). Making Confident Speaker Verification DecisionsWith Minimal Speech. *IEEE Transactions on Audio, Speech and Language Processing*, 18(6):1182–1192.

[Waibel et al., 1989] Waibel, A., Hanazawa, T., Hinton, G. E., Shikano, K., and Lang, K. J. (1989). Phoneme recognition using time-warping neural networks. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37(3):328–339.

[Wang et al., 2018] Wang, Q., Downey, C., Wan, L., Mansfield, P. A., and Moreno, I. L. (2018). Speaker Diarization with LSTM. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5239–5243.

[Yamaguchi et al., 2005] Yamaguchi, M., Yamashita, M., and Matsunaga, S. (2005). Spectral cross-correlation features for audio indexing of broadcast news and meetings. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 613–616.

[Zagoruyko and Komodakis, 2016] Zagoruyko, S. and Komodakis, N. (2016). Wide Residual Networks.

[Zelenák and Hernando, 2012] Zelenák, M. and Hernando, J. (2012). Speaker overlap detection with prosodic features for speaker diarisation. *Signal Processing, IET*, 6:798–804.

[Zhang et al., 2019] Zhang, A., Wang, Q., Zhu, Z., Paisley, J., and Wang, C. (2019). Fully Supervised Speaker Diarization. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6301–6305. IEEE.

[Zhang et al., 2014] Zhang, Y., Chuangsuwanich, E., and Glass, J. (2014). Extracting deep neural network bottleneck features using low-rank matrix factorization. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 185–189.