Efficient speaker diarization and low-latency speaker spotting

José María Patino Villar

SORBONNE UNIVERSITÉ

EURECOM
Sophia Antipolis

Efficient speaker diarization and low-latency speaker spotting

Dissertation

submitted to

Sorbonne Université

in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

Author:
José María PATINO VILLAR

Defended on the
24th of October 2019
before a committee composed of:

Thesis advisor	**Prof. Nicholas EVANS**, EURECOM, France
Reviewers	**Prof. Eduardo LLEIDA SOLANO**, Universidad de Zaragoza, Spain
	Prof. Sylvain MEIGNIER, Université du Mans, France
Examiners	**Prof. Isabel TRANCOSO**, Instituto Superior Técnico, Portugal
	Prof. Marc DACIER, EURECOM, France
Invited member	**Dr. Héctor DELGADO**, Nuance Communications, Spain

Abstract

Speaker diarization involves the detection of speakers within an audio stream and the intervals during which each speaker is active, i.e. the determination of 'who spoken when'. The task is closely linked to that of speaker recognition or detection, which involves the comparison of two, presumably single-speaker speech segments and the determination of whether or not they were uttered by the same speaker. Even if many practical applications require their combination, the two tasks are traditionally tackled independently from each other. The work presented in this thesis takes a similar approach, but also considers an application where speaker diarization and speaker recognition solutions are fused at their heart.

Both speaker diarization and recognition rely upon speaker modelling techniques, the most advanced of which exploit developments in deep learning. These techniques typically require training and optimisation using massive quantities of domain-matched training data. When such modelling techniques are applied to domain-mismatched data, performance can degrade substantially. When domain-matched data is scarce, or not available in sufficient quantities, then alternative, domain-robust speaker modelling techniques are needed. The work presented in this thesis exploits an approach to speaker modelling involving binary keys (BKs). BK modelling is inherently efficient and can operate without the need for external training data. Instead, it operates using test data alone. Novel contributions include: (i) a new approach to BK extraction based on multi-resolution spectral analysis; (ii) an approach to speaker change detection using BKs; (iii) the application of spectral clustering methods to BK speaker modelling; (iv) new fusion techniques that combine the benefits of both BK and deep learning based solutions to speaker diarization. All contributions lead to substantial improvements over baseline techniques and led to excellent results in 3 internationally competitive evaluations, including 2 best-ranked systems.

Other contributions include the combination of speaker diarization within a speaker detection task. The work relates to security applications and EURECOM's participation in the ANR ODESSA project. The new task, coined low latency speaker spotting (LLSS), involves the rapid detection of known speakers within multi-speaker audio streams. It

involves the re-thinking of online diarization and the manner by which diarization and detection sub-systems should best be combined. Novel contributions include: (i) a formal definition of the new LLSS task; (ii) protocols to support LLSS research using a publicly available database; (iii) LLSS solutions which combine online diarization with speaker detection; (iv) metrics for the assessment of LLSS performance as a function of speaker latency; (v) a selective cluster enrichment technique which fuses diarization and detection sub-systems at their heart. This work shows that the optimisation of speaker diarization solutions should not be performed using the traditional diarization error rate, but instead with metrics that better reflect the eventual application. Speaker diarization is an enabling technology and is almost never an application in its own right.

Acknowledgements

I would like to start by expressing my gratitude to my supervisor Prof. Nicholas Evans for giving me the chance to continue learning and developing my researching skills as a member of the Audio Security and Privacy group at EURECOM. Nick's support has been fundamental to finalise my doctoral studies both through his professional and personal support. While one may generally expect the former from an academic supervisor, Nick also excels in the latter, which I believe to be of an even greater importance. I will always be grateful for his help in navigating the ups and downs that happen during a PhD.

To my great friend and unofficial co-supervisor Dr. Héctor Delgado I am equally thankful. His guidance and professionalism, endless patience and thoroughness have taught and motivated me through these years. I cannot express enough my gratitude to him and I will cherish his lessons and friendship forever.

Besides them, I extend my appreciation to the members of my thesis jury committee for their generous time invested in the evaluation of this manuscript: Prof. Eduardo Lleida Solano, Prof. Sylvain Meignier, Prof. Isabel Trancoso, and Prof. Marc Dacier. Similarly, to the members of the ANR/SNF ODESSA project in which we had the chance to collaborate, in particular to Dr. Ruiqing Yin, Dr. Hervé Bredin, and Dr. Claude Barras, from LIMSI, and Dr. Alain Komaty, Dr. Pavel Korshunov, and Dr. Sébastien Marcel, from IDIAP.

Working in EURECOM has allowed me to be surrounded by great professionals from whom to learn, many of whom have also become friends. Without them my experience these years would have not been nearly the same, and to them I am also very thankful. From the audio group, to my good friends Pramod and Massimiliano, with whom I have shared countless discussions and laughs, as well as many pleasing trips. From the colleagues that are or were at EURECOM: Valeria, Pasquale, Antonio, Flavio, Giovanni, Chiara, Robin, Rajeev, Leela, Raj, Giacomo, Daniele, Enrico, Natacha, José Luis, Kalyana, Kurt, Emanuele, Lorenzo, Placido, or to the many friends outside from work: Helena (and Darío), Jemil, Joske, Aniko, Roberta, Alessandro, Nicole, Antoine, and those I must be forgetting. Thank you for these great years.

While living abroad is a great and enriching experience, one must also accept being apart from his closest friends. However, I enjoyed the recharging support received from

my lifelong pals during my visits back home: Eugenio, Ramón, Carlos, Javier, Emilio, Luis, Ricardo, Álvaro, Anthony and Germán. I am lucky to count you as friends.

To my partner Khawla, because your support and care beyond what is reasonable, your contagious happiness and optimism have kept me going and make my life so much prettier. You will always be the greatest discovery of this PhD, thank you for everything.

To conclude I want to say thank you to my family. To my parents, María and José María, for raising me in a life full of happiness, education and unconditional support. For all your sacrifices and headaches, thank you, the idea of making you proud is my greatest joy. To my sister Marta, for her trust and care. To my dear aunt Elvira, for her precious pampering. To my grandmother Dolores, for teaching me through the purest love and most beautiful wisdom.

Antibes, October 2019 ***Pepe***

Contents

List of Figures

List of Tables

Glossary

AHC Agglomerative hierarchical clustering

ASR Automatic speech recognition

ASV Automatic speaker verification

BIC Bayesian information criterion

BK Binary key

CQT Constant Q transform

CV Cumulative vector

DCF Detection cost function

DCT Discrete cosine transform

DER Diarization error rate

DFT Discrete Fourier transform

DHC Divisive hierarchical clustering

DL Deep learning

DNN Deep neural network

EER Equal error rate

EMB Triplet-loss neural embeddings

FA False accept/alarm

FE Feature extraction

FFT Fast Fourier Transform

FR False reject

GMM Gaussian mixture model

ICMC IIR-CQT Mel-frequency cepstral coefficient

IIR Infinite impulse response

KBM Binary key background model

LLSS Low-latency speaker spotting

LSTM Long short-term memory

MD Miss detection

MFCC Mel-frequency cepstral coefficient

PLDA Probabilistic linear discriminant analysis

SC Spectral clustering

SCD Speaker change detection

SCE Selective cluster enrichment

SD Speaker diarization

SER Speaker error rate

STFT Short-time Fourier transform

UBM Universal background model

VAD Voice activity detection

WCSS Within-cluster sum of squares

Chapter 1

Introduction

Data quantities at the disposal of researchers have been increasing massively during the past two decades. This is due to a variety of factors that include steady hardware developments and improved, almost-ubiquitous connectivity, enabling for the continuous capture and storage of all sorts of information. The benefits of this data to society lie in what can be learned from it. Research in this area has led to the birth of a new field of research known as data science and renewed efforts to develop the mathematical tools required for efficient and domain robust pattern recognition. Much of the research in this area involves machine and deep learning (DL).

This thesis concerns *speech* data, which is not an exception in terms of its increased abundance in recent years. Following the explotion in the ubiquity of handheld devices equipped with microphones, speech is now acquired more easily than ever, and by means that are non-intrusive to the user experience. The increasing number of always-listening connected devices through the *internet of things* also contributes to continuous speech acquisition that is of benefit in a great number of applications. In parallel, wireless, ubiquitous connectivity enables for cloud-based systems to perform all kinds of increasingly demanding processing operations remotely rather than locally, bringing handheld device hardware requirements to a bare minimum. These advances in hardware and connectivity provide, today more than ever, an ideal environment for the real usability of speech processing technologies.

The development of tools capable of processing and exploiting the content of conversational speech is motivated by the fundamental importance of speech in society: speech constitutes one of the most natural means of human interaction, and the information conveyed in speech is especially rich and diverse. Perhaps the most obvious example of speech processing tasks is that of automatic speech recognition (ASR). Estimating the

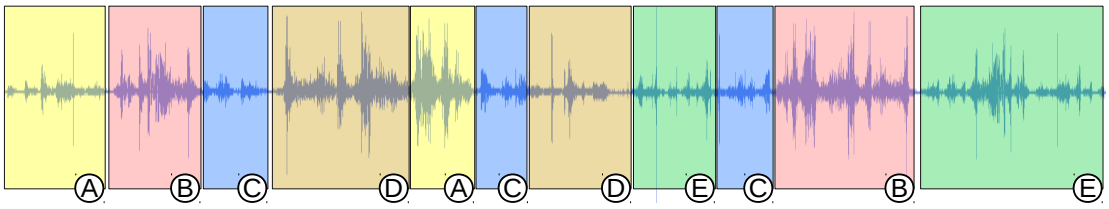


Figure 1.1: The task of SD attempts to segment an audio stream into speaker homogeneous intervals and to group together same-speaker segments with a common speaker label (each color and letter represents a different speaker).

words uttered in an audio stream may be of importance to many different applications, for example, the indexing and cataloging of spoken content. These applications among many others can be a component of natural language understanding, and the development of human-machine interaction and artificial intelligence. Less obvious applications that, nonetheless, represent established fields of research are those comprising the estimation of behavioral content, such as intent or emotion. Last but not least, speech signals also communicate the speaker identity. Human-to-human communication naturally exploits this information: our brain is capable of estimating age and gender, and differentiates among speaker identities when listening to even very short amounts of speech. Estimation of the same information by *machines* is considerably more challenging. Application related to voice biometrics, namely the estimation of speaker identities in speech data, is the main focus of this dissertation for two particular applications: first and foremost that of speaker diarization (SD) [1, 2] and, second, the integration of diarization with speaker recognition [3, 4].

Speaker diarization is commonly referred to as the task of determining '*who spoke when*' in a multi-speaker audio stream, a conceptual example of which is illustrated in Figure 1.1. As an enabling technology, SD aims to enrich the transcription of audio datasets with speaker labels. These can be useful for numerous tasks including information indexing, tools for the hearing impaired, enriched interaction with always-listening devices, or applications that involve the tracking of certain speakers. The datasets used in the study of SD reflect to some extent the expected acoustic domains of these use cases. Telephony databases comprising conversations mostly between two speakers are useful for security and commercial applications, e.g. the (re)detection and tracking of known fraudsters, or the verification of user identities in call-center applications. SD in applications involving broadcast news radio and television content bring more significant challenges related to variations in acoustic variability. SD in meeting domain application typically bring even more challenging problems related to unstructured, spontaneous conversational speech.

The speaker labelling task is more often than not performed without any a priori

information about the audio content, e.g. the nature of the speech content and/or presence of acoustic nuisances such as music, clapping, background conversations or noise. In addition, the number of speakers is often unknown, and the length of the speech segments related to the conversations can go from brief interventions, to overlapping interruptions, and even single-speaker monologues.

The complexity of the SD task normally leads to a similarly complex SD system pipelines which typically include a number of independent modules such as voice activity detection (VAD), feature extraction, speaker segmentation and modelling, clustering, and resegmentation. Research as regards some of these modules often draws from other more established fields, e.g. speaker recognition, in which DL based approaches have recently overtaken the previous state-of-the-art [5]. The most substantial improvements in performance reported in the literature typically leverage vast amounts of training data [6, 7, 8] often result in inefficient SD solutions that work only well in matched-domain conditions.

1.1 Domain robust and efficient speaker diarization

What may be an asset for training data-dependent techniques, i.e., the capability to leverage massive quantities of in-domain labelled data, may also be a drawback in the context of unseen domains. Domain-variability and mismatch is a very common and complex issue in the deployment of real systems and SD is no exception. The performance of state-of-the-art solutions is usually reported for selected acoustic scenarios related to narrow use cases for which training data is abundant. However, for many other use cases matched training data is scarce, leading to degraded performance. Even if domain-matched training data were to become available, typical SD solutions then require retraining or adaptation and re-optimisation. This can be highly computationally demanding and inefficient.

The focus of the work exposed in the first part of this thesis is thus motivated by the need for domain-robust, efficient SD solutions that function reliably even without domain-dependent training data. The principal contributions of the work presented involve an approach to improve both efficiency and domain robustness. The solution lies in the form of binary key (BK) speaker modelling [9]. While originally proposed for the task of speaker recognition, BK-based approaches to SD have also been proposed [10, 11] and result in an unsupervised, easily tunable, and computationally light solution. BK-based SD does not need to leverage external training data but rather exploits acoustic information extracted from the audio data itself. Under such a scenario there are no mismatch problems (there is no external training data), making BK-based solutions

also particularly well suited to rapid deployment in cases where hand-labelled resources may be scarce. Even so, BK-based solutions tend not to perform as well as competing alternatives. This thesis shows how BK-based solutions can be improved to match and even out-perform the very best alternative techniques.

1.2 Low-latency speaker spotting

A second line of research is also pursued in this thesis. It relates to a practical application of SD. As an *enabling technology*, it is arguable that research in SD lacks a focus on any specific application whereas, in practice, solutions must necessarily be optimised with the final application in mind. Instead, SD is often tuned to optimise a diarization-based metric that all but ignores how the SD algorithm should be tuned for a specific application. Such metrics tend to promote the consistent labelling of dominant speakers while not penalising mistakes related to occasional, less dominant speakers. Diarization, is thus effectively optimised as an application in its own right.

The optimisation of SD for a specific application (here related to security) is one of the research goals of the *Online Diarization Enhanced by recent Speaker identification and Sequential learning Approaches* (ODESSA) project, a French-Swiss research collaboration funded by the French Agence National de la Recherche (ANR) and the Fonds National Suisse de la recherche scientifique (FNS). In particular, ODESSA puts the focus on real use cases related to security and low-latency text-independent automatic speaker verification (ASV) in the context of multi-speaker audio streams, e.g. the rapid detection of blacklisted, known speakers under surveillance. ASV involves the comparison of speaker identities in a pair of utterances, i.e. determining whether they were uttered by the same or different speaker(s). While nowadays a relatively mature technology in the case of controlled, long-utterance scenarios, ASV performance still degrades when constrained to operate short-utterances, or when evaluated with audio segments containing more than a single speaker. *Online* SD is then an obvious means of improving ASV performance. To the best of the author's knowledge, the proposed scenario, which requires the integration and joint assessment of online SD and ASV systems, as illustrated in Figure 1.2, has never been considered before. This new task, coined *low-latency speaker spotting* (LLSS), constitutes the second major contribution of this thesis.

1.3 Contributions and thesis outline

The structure of this thesis is illustrated in Figure 1.3. It is divided into two main parts, each of which spans one of the main themes: domain-robust, efficient BK-based

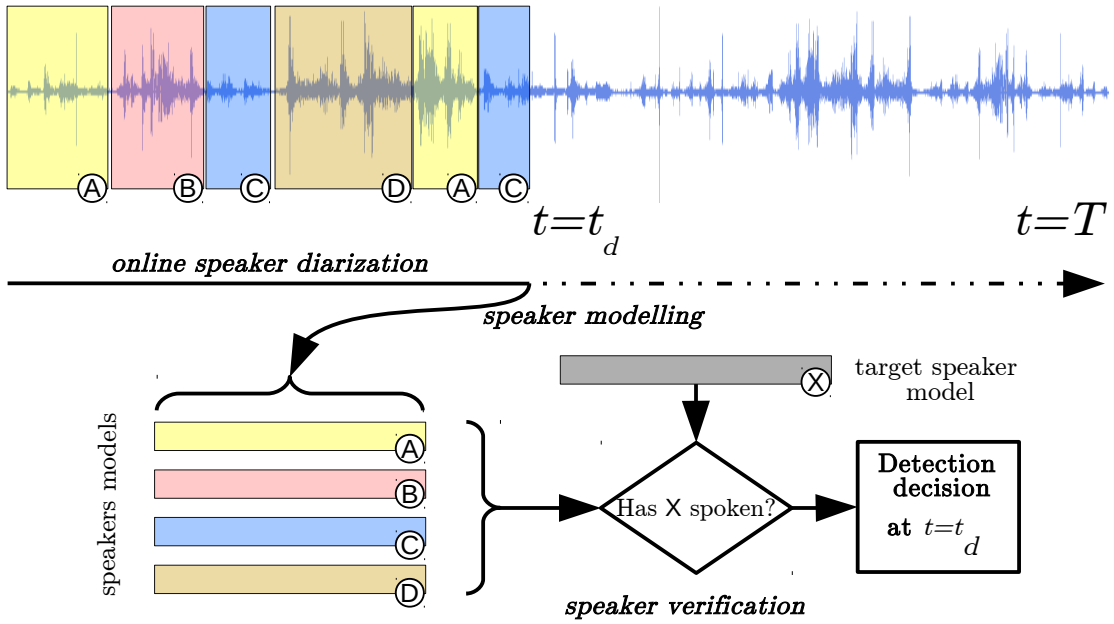


Figure 1.2: Illustrated here is the use case that motivates the second line of research pursued during this thesis. Online SD is applied over an incoming audio stream to estimate the speaker identities at time $t = t_d$. Audio in the resulting speaker clusters are turned into speakers-discriminative representations by means of speaker modelling, and compared in real time with the speaker model of a previously enrolled individual. An adequate integration of these SD and recognition systems should allow for the low-latency speaker detection of a target speaker.

approaches to SD, and LLSS. Contributions reported in Part I include a number of enhancements to a BK-based approach to SD that delivers state-of-the-art performance. This technique offers an efficient and domain-independent approach to SD that avoids the use of external training data. As such, it can be applied readily to unseen scenarios or to new data domains without the need for heavy computational processing or adaptation. This characteristic does, nonetheless, place strict constraints upon the lines of research in that new developments must also remain training data independent. The thesis reports numerous enhancement to various modules of the baseline system that respect this constraint. The aim of the work is to reduce the gap between BK-based and more traditional approaches to SD. The contributions include novel enhancements to feature extraction, speaker change detection, clustering, and fusion techniques. On account of the timing between the development of the work presented here and the revitalisation of research in SD, experiments are reported in the context of recent evaluations and related, standard datasets. These include broadcast news datasets [12, 13] and more unusual scenarios in the form of a multi-domain dataset [14]. These frameworks allowed for the contributions presented in this thesis to be evaluated against the competitive solutions

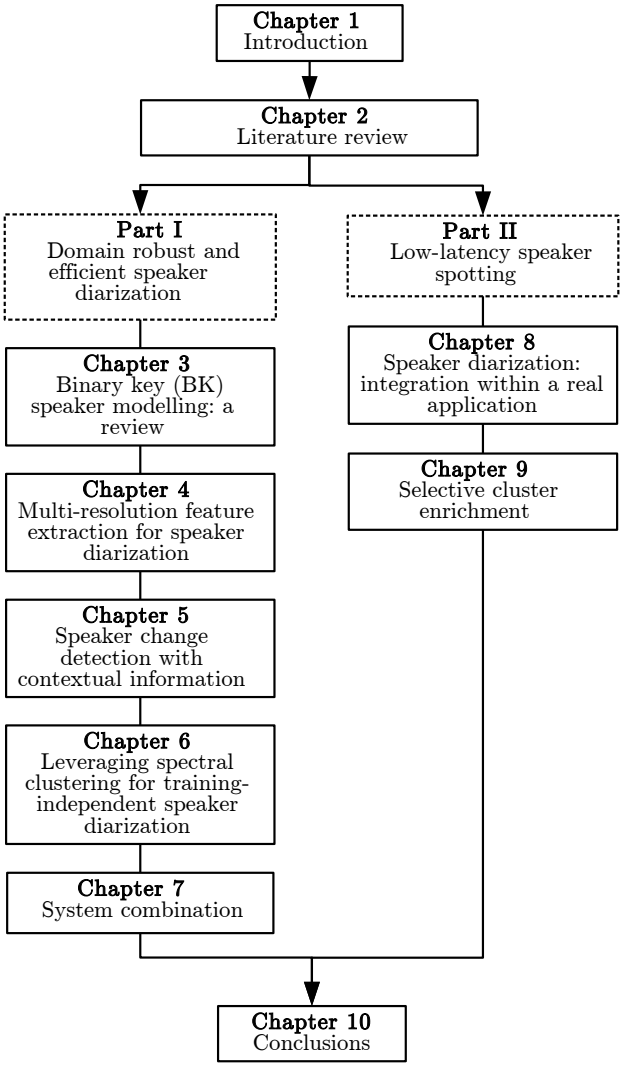


Figure 1.3: The outline of the content in this manuscript.

of leading international research teams in the field.

Systems were submitted to three international SD evaluations achieving 1st place in two of them:

- 1st position in the Albayzin 2016 Speaker Diarization Evaluation
- 1st position in the Albayzin 2018 Speaker Diarization Challenge (*open-set* track)
- 2nd position in the Albayzin 2018 Speaker Diarization Challenge (*closed-set* track)

Contributions in Part II of this thesis all relate to LLSS. The thesis introduces the first framework for the integration of an online SD system with speaker verification. A

solution for online diarization, a notoriously challenging problem [15, 16, 17], is based on an i-vector framework and performed by means of a greedy, online sequential clustering algorithm. Different baseline detection systems are proposed and tested with performance metrics traditionally used in speaker recognition, but modified to consider latency with regard to *speaker* content rather than *absolute* latencies. This task is coined as *low-latency speaker spotting*. To the best of the author's knowledge this is an entirely new task which provides a framework for low-latency speaker verification rather than online diarization. Contributions include a formal definition and new protocols to support its exploration using an existing, public dataset. In addition to reporting the very first solutions, other contributions include an approach to better guide the online clustering algorithm. This work in particular serves as one example of how SD research must reflect the final application. Consideration of the full application pipeline and the fusion of SD with speaker recognition leads to a hybrid system. This only serves to show further the importance of the eventual application in the case of SD research.

In addition to these two parts, Chapter 2 presents a review of the state-of-the-art in SD and modelling. Conclusions spanning the contributions in both parts are presented in Chapter 10. The following outlines the structure and contributions in each chapter.

Part I. Domain robust and efficient SD

The first part of this thesis focuses on the problem of domain-robustness and efficiency. Both requirements are met with a so-called binary (BK) approach to SD. Contributions in Part I relate to improvements to various aspects of the BK SD pipeline. These all aim to reduce the gap between training-dependent and training-independent solutions to SD.

Chapter 3

Chapter 3 presents a review of the original BK algorithm for the task of speaker recognition, as well as its adaptation through later work to SD. The application of BK modelling to the two applications leads to differences in the training methods for the binary key background model (KBM). The emphasis is on explaining how traditional acoustic features are transformed into the binary domain in order to derive segmental level, speaker-discriminative representations. In addition, metrics used to evaluate similarities in the BK space are discussed, as well as recent improvements to the original algorithm proposed by other authors. The baseline SD system used in this thesis then is defined. It is to this baseline system that subsequent enhancements are applied.

Chapter 4

The work in Chapter 4 presents the first application of multi-resolution spectral analysis to BK speaker modelling. Recent works in speech processing tasks have shown benefits in performance achieved by means of constant Q transform spectral analysis. Here it is applied for the first time to the task of SD. Contributions include (i) an analysis of its impact on the speaker-discriminative capacity of BK-based solutions to both speaker recognition and SD and (ii) optimisations to front-end processing. Results show substantial improvements, which led to the proposed system with infinite impulse response, constant-Q (IIR-CQT) Mel-frequency cepstral coefficients (ICMC) being awarded 1st place in the Albayzin 2016 Speaker Diarization Evaluation.

Part of the work presented in this chapter was published in:

- **J. Patino**, H. Delgado, N. Evans, and X. Anguera, "EURECOM submission to the Albayzin 2016 Speaker Diarization Evaluation," in *Proc. IberSPEECH*, Lisbon, Portugal, October 2016

which shows that the proposed system delivers a relative improvement of 14% DER over the baseline.

Chapter 5

Following the focus on front-end processing in Chapter 4, Chapter 5 describes a new, explicit speaker change detection (SCD) mechanism based entirely on BK speaker modelling. This is an alternative to the straight-forward homogeneous segmentation approach of the baseline SD system. Other contributions include the comparison of two methods to KBM composition: the baseline method is compared to a novel algorithm that emphasizes the relevance of local, contextual information in the surroundings of a hypothesised speaker change point. The proposed system is compared to a Bayesian Information Criterion (BIC) solution. In keeping with other, similar work reported in the literature, this work was performed on the ETAPE database.

Part of the work presented in this chapter was published in:

- **J. Patino**, H. Delgado, and N. Evans, "Speaker Change Detection Using Binary Key Modelling with Contextual Information," in *Proc. International Conference on Statistical Language and Speech Processing*, Le Mans, France, October 2017

which shows that the novel, local-context KBM composition method leads to a relative increase in average segment coverage of 17.4% compared to the baseline.

Chapter 6

Chapter 6 switches focus to clustering. Contributions include the first application of spectral clustering (SC) to the BK-based SD system. SC is based on the eigenvector decomposition of an affinity matrix. The work shows how it can be tailored to the needs of BK-based diarization so that: (i) it can be applied as a means of unsupervised dimensionality reduction in the form of chunked eigenvectors upon which partitional clustering is explored, (ii) it can be exploited in order to determine the number of speakers with an eigengap-based criterion which may be coupled with the AHC algorithm, and (iii) it allows for the reliable detection of single-speaker sessions. All these experiments were performed in the context of the first DIHARD challenge, which was based upon a new multi-domain SD dataset composed of audio recordings collected in challenging scenarios such as court rooms or clinical interviews concerting pathological disorders. This work shows that BK-based SD solutions compare favourably to DL-based solutions, while vastly outperforming those in terms of computational efficiency.

Part of the work presented in this chapter was published in:

- **J. Patino**, H. Delgado, and N. Evans, "The EURECOM subsmission to the first DIHARD challenge," in *Proc. INTERSPEECH*, Hyderabad, India, September 2018

which shows that the application of SC-derived methods to BK-based SD lead to relative improvements in the order of 40% DER over the baseline.

Chapter 7

Despite its appeal in terms of domain robustness and computational efficiency, compared to results obtained by DL based approaches, the BK-based system falls slightly short. In contrast to the BK-based system, DL-based approaches leverage large quantities of external training data, the use of which can deliver better performance in the case that training data matches the domain of the test/evaluation data. Since the BK-based SD system does not use external training data, it is domain-independent, meaning that it can be applied readily in new data domains without costly adaptation or retraining. At the same time, it remains highly computationally efficient. Even if BK and DL-based systems are designed with different operational criteria in mind, it is of interest to determine their complementarity; are they using the same cues and can they be combined to improve performance? The work reported in Chapter 7 explores two system combination techniques in order to merge different BK systems and/or their outputs. The work was performed with the BK-based system and two different neural embedding-based approaches. Results of this work, undertaken in collaboration with partners from the ODESSA project, are reported in the context of the Albayzin 2018 Speaker

Diarization Challenge, the training conditions of which (*open-* and *closed-set*) provided an ideal scenario for the exploration of different system fusion approaches. The ODESSA submission employing a similarity-matrix level approach to SD fusion was awarded 1st position for the open-set condition. The submission to the closed-set condition based on a hypothesis-level fusion was awarded 2nd position. These results show that the BK-based SD system, in addition to its computational efficiency and domain robustness, is complementary to neural embedding systems.

Part of the work presented in this chapter was published in:

- **J. Patino**, H. Delgado, R. Yin, H. Bredin, C. Barras and N. Evans, "ODESSA at the Albayzin Speaker Diarization Challenge 2018," in *Proc. IberSPEECH*, Barcelona, Spain, November 2018

which shows that compared to the performance of single systems the combination of BK-based speaker-discriminative representations and neural embeddings delivers a relative improvement of 8% DER by means of a similarity-matrix level approach to fusion, and 13% DER when fusion is applied at the hypothesis-level.

Part II. Low-latency speaker spotting

The second part of this thesis explores SD in terms of a real application, namely speaker detection task referred to as low-latency speaker spotting (LLSS). The low-latency requirements calls for the joint application of online SD and speaker detection methods, which are explored within a new evaluation framework.

Chapter 8

Chapter 8 introduces the newly coined task of low-latency speaker spotting (LLSS). It provides a formal definition of the task, i.e. the detection, as soon as possible, of known speakers within multi-speaker audio streams. The differences with regard to traditional SD and recognition are described. New metrics that consider latency are proposed and protocols for the assessment of the task using a publicly available dataset are presented. The LLSS solution combines an online diarization system based upon the sequential clustering of i-vectors with different speaker recognition systems based on GMM-UBM, i-vectors, and neural embeddings techniques. This work, also undertaken in collaboration with partners of the ODESSA project, highlights the challenging nature of the task and the need to consider SD in the context of the final application; it is an enabling technology and rarely an application in its own right.

Part of the work presented in this chapter was published in:

- **J. Patino**, R. Yin, H. Delgado, H. Bredin, A. Komaty, G. Wisniewski, C. Barras, N. Evans, and S. Marcel, "Low-latency speaker spotting with online diarization and detection," in *Proc. Speaker Odyssey*, Les Sables d'Olonne, France, June 2018

which presents the LLSS task and the results obtained by the first proposed solution.

Chapter 9

The work presented in Chapter 9 breaks with the strategy of separate SD and speaker detection systems. It takes a first step to move beyond the traditional definition of SD and more towards an application-related, joint optimisation. It presents a new approach to incorporate known speaker models into the heart of the online clustering algorithm. Further contributions include a so-called selective cluster enrichment (SCE) process which guides the online LLSS clustering algorithm towards purer target-model related clusters in a manner that prioritizes ASV performance over SD performance.

Part of the work presented in this chapter was published in:

- **J. Patino**, H. Delgado, and N. Evans, "Enhanced low-latency speaker spotting using selective cluster enrichment," in *Proc. International Conference of the Biometrics Special Interest Group (BIOSIG)*, Darmstadt, Germany, October 2018

which shows the benefit of the cluster enrichment approach with a relative increase of 9% EER for speaker latencies of 15s over the baseline.

Publications

Discussed in this manuscript

1. **Jose Patino**, Héctor Delgado, Nicholas Evans and Xavier Anguera, "**EURECOM submission to the Albayzin 2016 Speaker Diarization Evaluation**" in *Proc. IberSPEECH*, Lisbon, Portugal. November 2016.

2. **Jose Patino**, Héctor Delgado and Nicholas Evans, "**Speaker change detection using binary key modelling with contextual information**", in *Proc. International Conference on Statistical Language and Speech Processing (SLSP)*. Le Mans, France. October 2017.

3. **Jose Patino**, Ruiqing Yin, Héctor Delgado, Hervé Bredin, Alain Komaty, Guillaume Wisniewski, Claude Barras, Nicholas Evans and Sébastien Marcel, "**Low-latency speaker spotting with online diarization and detection**", in *Proc. Speaker Odyssey*. Les Sables d'Olonne, France. June 2018.

4. **Jose Patino**, Héctor Delgado and Nicholas Evans, "**The EURECOM submission to the first DIHARD Challenge**", in *Proc. INTERSPEECH 2018*. Hyderabad, India. September 2018.

5. **Jose Patino**, Héctor Delgado and Nicholas Evans, "**Enhanced low-latency speaker spotting using selective cluster enrichment**", in *Proc. International Conference of the Biometrics Special Interest Group (BIOSIG)*. Darmstadt, Germany. September 2018.

6. **Jose Patino**, Héctor Delgado, Ruiqing Yin, Hervé Bredin, Claude Barras and Nicholas Evans, "**ODESSA at Albayzin Speaker Diarization Challenge 2018**" in *Proc. IberSPEECH*. Barcelona, Spain. November 2018.

Other work

7. Benjamin Maurice, Hervé Bredin, Ruiqing Yin, **Jose Patino**, Héctor Delgado, Claude Barras, Nicholas Evans and Camille Guinaudeau, "**ODESSA/PLUMCOT at Albayzin Multimodal Diarization Challenge 2018**" in *Proc. IberSPEECH*. Barcelona, Spain. November 2018.

8. Andreas Nautchs, **Jose Patino**, Amos Treiber, Themos Stafylakis, Petr Mizera, Massimiliano Todisco, Thomas Schneider and Nicholas Evans, "**Privacy-Preserving Speaker Recognition with Cohort Score Normalisation**", in *Proc. INTERSPEECH*. Graz, Austria. September 2019.

9. Kong Aik Lee et al., "**I4U Submission to NIST SRE 2018: Leveraging from a Decade of Shared Experiences**", in *Proc. INTERSPEECH*. Graz, Austria. September 2019.

Chapter 2

Literature review

This chapter provides an overview of the literature of relevance to the work developed in this thesis. Alternatives to the processing modules that compose a traditional speaker diarization (SD) pipeline are presented. These include acoustic feature extraction, voice activity detection, speaker segmentation, speaker modelling clustering and resegmentation methods. Finally, SD and speaker recognition evaluation methods are introduced. Further, detailed read can be found in widely cited survey articles for SD [1, 2] and speaker recognition [3, 4].

2.1 Acoustic feature extraction

Feature extraction is a basic processing step in pattern recognition applications. This step is traditionally performed by means of handcrafted techniques that attempt to emphasize information that is relevant to a final application, providing with a computationally lighter, and hopefully more discriminative data representation. Feature extraction related to voice biometrics, which concern the work in this thesis, should generate, in an ideal scenario, features which, quoting [18] and [3], (i) maximize & minimize, respectively between- & intra-speaker variabilities, (ii) provide robustness to noise and distortion, as well as physiological factors such as aging or health, (iii) can be easily both found and captured from natural speech content, and (iv) cannot be easily impersonated. However, a *perfect* feature extraction that satisfies all these requisites does not yet exist. In practice, state-of-the-art performance in voice biometrics has mostly been developed by using acoustic features widely popular in speech procesing tasks, which are the Mel-frequency Cepstral coefficients (MFCCs) [19]. Alternatives are based on different approaches such as *linear prediction* [20] in the form of linear predictive cepstral coefficients (LPCCs) [21], linear spectral frequencies (LSFs) [21] or perceptual linear

prediction (PLP) coefficients [22]. However, MFCCs remain the option of choice for a large parte of the voice biometrics community. MFCCs are motivated by human hearing perception, and fall within the category of *short-term spectral features*, as they are extracted from speech frames that contain between 20 and 30 ms of raw speech data. Following the notation introduced in [3] for a physical interpretation of speech features, short-term spectral features correspond with descriptors of the *spectral envelope*, acoustically related to the *timbre* of the voice and the resonances generated by the human vocal tract.

In continuing with the physical interpretation of acoustic features, different front-ends for SD have been developed by leveraging prosodic and spectro-temporal information as an alternative to MFCCs. Prosodic features are extracted from speech that is framed in lengths that span from hundreds to thousands of miliseconds, and put the focus on speech information such as the pitch or the energy of the signal. In [23] the authors prove that prosodic features can be of benefit to diarization performance when combined with MFCCs, but not as a stand-alone front-end. Another example of prosodic features is that of [24] where the use of jitter and shimmer, speech quality measures used to detect voice pathologies [25] and speaking styles [26], generated a slight improvement in diarization performance but, once again, when coupled with MFCCs.

Higher in the abstraction level, so-called high-level features allow to capture content that include phones, words, or articulatory speech features. Whilst these kind of features have led to some benefit when used in speaker recognition [27, 28], their study is not common in SD. However recent work such as that in [29] present promising results by leveraging higher level speaker traits such as age, gender, or voice likability.

Finally, deep neural networks (DNNs) are also employed for feature extraction by means of bottleneck features. While initially integrated in speaker recognition systems [30, 31], works in the literature have recently proposed their use in speaker clustering [32] and diarization [33, 34, 35].

2.2 Voice activity detection

Voice activity detection (VAD), also referred to as speech activity detection (SAD), performs the task of segmenting an audio stream into speech and non-speech content. Non-speech content refers to silence, background noises coming from the environment, e.g. clatter, clapping, laughter or music. This kind of content is considered non-informative for the tasks related to voice biometrics. In speaker recognition, non-speech content is assumed to affect the robustness of the solutions. The same applies for SD where, however, a following clustering step may be widely affected by nuisances in the form of

non-speech.

A precise VAD is consequently of great importance for the SD task and its performance. The value of detecting speech content and not miss-classifying it as non-speech is obvious as (i) performance is decremented by default in an unrecoverable manner, and (ii) the missed-speech data will force whichever speaker modelling technique next in the SD pipeline to operate on fewer speech data. This leads to diminished robustness and a clustering more prone to errors. A similar logic applies to the incorrect classification of non-speech as speech data. Non-speech audio segments will contaminate the clustering process by misguiding the merging/splitting steps involved, and deteriorating results.

In terms of algorithms, two main kind of approaches are considered when it comes to VAD. A first one is that based on energy levels present in the signal. These kind of algorithms are considerably accurate when it comes to controlled scenarios in which energy levels in the signal are maintained within consistent ranges. This is a somewhat acceptable constraint in some speaker recognition scenarios such as telephony [3], where speakers are expected to be continuously close to the microphone and background noises are limited in variety. Such a limitation cannot, however, cannot be guaranteed in SD audio files, which are characterised by being considerably lengthier than their speaker recognition counterparts in a variety of domains, e.g. broadcast news, meeting environments. A much wider variability is thus present, limiting their applicability in the field.

Most approaches to VAD are consequently not energy-based. On the contrary, thanks to the relatively easy labelling task of this 2-class problem, large amounts of training date are readily available, motivating the success of model-based approaches to VAD, i.e. a model is previously trained on background data in order to discern between speech and non-speech content. Traditional methods have relied on Gaussian mixture models (GMMs) [36, 37], with proposed modifications allowing them to adapt iteratively over test data [38]. These GMM models are usually trained to represent speech and non-speech acoustic classes, although in the presence of richer labelling in the training data which includes sub-classes of non-speech, e.g. music or noise, extra classes may also be considered. The relationship to the acoustic features is assigned by means of Viterbi decoding [39].

Model-based VAD methods do however suffer when facing domain mismatches. Robustness against such mismatches has become increasingly available thanks to many modern methods developed to leverage developments in deep learning (DL), achieving state-of-the-art performance. Many different architectures have been proposed, including variations of feed-forward DNNs [40], convolutional neural networks (CNNs) [41, 42], or

long-short term memory (LSTM) neural networks [43, 44]. A VAD system following the approach introduced in [44] is used in some of the work reported in Chapters 7, 8 and 9.

2.3 Segmentation and speaker change detection

Following the pipeline of the traditional SD system comes the speaker segmentation and/or speaker change detection (SCD) module. Given that SD operates upon multi-speaker audio streams, it seems reasonable to perform some sort of pre-clustering processing that allows to separate speech segments from different speakers into homogeneous segments. Considering that a whole chapter of this thesis relates about this task (see Chapter 5) this section is deliberately brief in the justification and implications of SCD. However, a few methods of interest for the read of Chapter 5 are discussed here.

Approaches to SCD may be divided with regard to their dependence to external training data. The simplest approaches to SCD rely on *implicit* methods to segmentation. Examples of such approaches would be a VAD-based segmentation in which the output of the VAD system is assumed to separate speech segments into single-speaker speech fragments. Such an approach may be an option in conversational speech guided by a very clear structure in which interruptions between speakers do not happen, and speech turns are respected. A second implicit approach to SCD is that of segmenting the speech segments derived from the VAD system into shorter, homogeneous speech segments with a certain overlap. This approach relies on sufficiently short speech segments to have been uttered by a single speaker. Whilst errors are very likely to occur by operating this way, it is hoped that a following clustering and/or resegmentation step(s) will correct them. More elaborated training-independent methods perform an *explicit* SCD relying on the computation of distance metrics. These measure the similarity between the speech content contained in two adjacent sliding windows. Speaker change points are hypothesised when the resulting score surpasses a certain empirically optimised threshold. A few examples follow:

Bayesian information criterion (BIC): The BIC was originally introduced in [45] as a metric to estimate the representativity of a model over the data on which it has been trained, by means of a likelihood criterion that is penalised by complexity. Given a set of N data points $\mathcal{X}$, and being M a model fitted to represent $\mathcal{X}$, then the BIC of M is defined as:

$$BIC(M) = log(\mathcal{L}(\mathcal{X}, M)) - \lambda \frac{1}{2} \#(M) log(N) \tag{2.1}$$

where $log(\mathcal{L}(\mathcal{X}, M))$ is the log-likelihood of the data $\mathcal{X}$ given the model M, λ is a data-dependent penalty term, and $\#(M)$ denotes the number of parameters in the model M.

The use of BIC for SCD was proposed in [46], where the task is regarded as an *hypothesis test* regarding the content of the adjacent sliding windows (herein represented by $\mathcal{X}_i$ and $\mathcal{X}_j$) being analysed. In this context, the hypothesis H_0 denotes both speech segments belong to a unique speaker and there is not a change point between $\mathcal{X}_i$ and $\mathcal{X}_j$. Alternatively, H_1 indicates both speech segments belong to different speakers, indicating a possible change between speakers. A speaker change point is thus hypothesised using the increment ΔBIC between $BIC(H_1)$ and $BIC(H_0)$ so that:

$$\Delta BIC = BIC(H_1) - BIC(H_0) = R(i,j) - \lambda P \tag{2.2}$$

where $R(i,j)$ denotes the difference between the log-likelihoods of the two hypothesis and P is the complexity penalty term. Full details may be found in [46].

A common approach is using GMMs to model the hypothesised speech segments, turning Equation 2.2 into:

$$\Delta BIC = log(\mathcal{L}(\mathcal{X}, M)) - (log(\mathcal{L}(\mathcal{X}_i, M_i) + log(\mathcal{L}(\mathcal{X}_j, M_j)) - \lambda \Delta \#(i,j) log(N) \tag{2.3}$$

where $BIC(H_1)$ corresponds to the log-likelihood of $\mathcal{X}$ being modelled by M, $BIC(H_0)$ with the summed log-likelihoods of two different GMMs M_i and M_j fitted to their respective window contents $\mathcal{X}_i$ and $\mathcal{X}_j$, and $\Delta\#(i,j)$ with the difference in number of free parameters between M and $M_i + M_j$. This algorithm was used as baseline for the SCD work reported in Chapter 5.

Generalised likelihood ratio (GLR): GLR is another popular alternative to metric-based SCD used in the literature [47, 48, 49] defined as:

$$GLR(\frac{H_0}{H_1}) = \frac{\mathcal{L}(\mathcal{X}, M)}{\mathcal{L}(\mathcal{X}_i, M_i) + \mathcal{L}(\mathcal{X}_j, M_j)} \tag{2.4}$$

which, following the notation used for the BIC, denotes a simpler hypothesis test that is not penalised by complexity.

Symmetric Kullback-Leibler (KL2): Used for speaker segmentation ([50]), the KL2 may be used for a similar hypothesis as in the previous two methods. It is defined as,

$$D_{\mathrm{KL2}}(P \parallel Q) = D_{\mathrm{KL}}(P \parallel Q) + D_{\mathrm{KL}}(Q \parallel P), \tag{2.5}$$

It is a symmetrized version of the standard KL Divergence (KL). Considering two

distributions P and Q, the KL divergence is defined as:

$$D_{\mathrm{KL}}(P \parallel Q) = \frac{1}{2}\left(\mathrm{tr}\left(\Sigma_Q^{-1}\Sigma_P\right) + (\mu_Q - \mu_P)^{\mathsf{T}}\Sigma_Q^{-1}(\mu_Q - \mu_P) - k + \ln\left(\frac{\det\Sigma_Q}{\det\Sigma_P}\right)\right) \tag{2.6}$$

where Σ_P, μ_P, Σ_Q and μ_Q represent the covariance matrices and mean vectors of the distributions P and Q which model data of dimension k.

Alternatively, a different set of approaches to SCD are these which use training data to segment the audio content. The mechanic here is similar to that of the training-independent methods: the content in adjacent windows is represented in a speaker discriminative fashion by means of some previously trained model capable of discriminating between speakers. The leveraging of training data leads to expectedly more robust SCD hypothesis given the right match between training and testing domains. Traditional approaches have used GMMs [51], or have leveraged techniques in speaker modelling such as a universal background model (UBM) [52], or the i-vector paradigm [53, 54]. The recent developments in DL have also reached this field, allowing for the surge of different methods based on neural networks. The authors in [55] tested a confidence labelling approach in order to get a DNN to classify speech windows as likely to contain a speaker change or not, operating directly on MFCCs. Alternatively, the authors in [56] employed a CNN and a similar labelling approach to perform SCD while operating upon spectrograms. In [57] speech segments are projected into a space modelled by means of the triplet-loss paradigm [58] using an LSTM. In this resulting space the new representations are separable by means of the Euclidean distance, in a technique later developed in [59].

2.4 Speaker modelling

Next in the SD pipeline is one of the most important modules, that which involves speaker modelling. Following the VAD and segmentation modules, the system should now operate upon *ideally* speaker-homogeneous speech segments to obtain speaker-discriminative representations. This section focuses on the techniques that have led the development in speaker modelling for both speaker recognition and diarization over the last years in the literature. In particular, and due to their relevance to some works reported in this thesis, it reviews the traditional Gaussian mixture model-Universal Background Model (GMM-UBM) paradigm, followed by i-vector based methods, and concludes with approaches based on neural networks and DL.

GMM-UBM

GMMs are generative models commonly employed in speaker recognition and diarization since their proposal in [60] followed by developments by the same authors in [61, 62], thanks to their capacity to fit the variability in a speech signal. GMMs are defined by a fixed number of Gaussian components K combined by means of a weighting factor. A GMM model λ may thus be characterised following its probability density function for a given observation x so that:

$$p(x|\lambda) = \sum_{k=1}^{K} w_k \mathcal{N}(x|\mu_k, \Sigma_k) \tag{2.7}$$

where w_k is the prior probability used as a mixing weight of each Gaussian component so that $\sum_{k=1}^{K} w_k = 1$, and μ_k and Σ_k are, respectively, the mean and covariance matrix of each k Gaussian component. Consequently, given a set of N acoustic feature vectors $\mathcal{X} = x_1, ...,$ contained in a speech segment, its log-likelihood given a GMM model λ is:

$$log(p(\mathcal{X}|\lambda)) = \sum_{n=1}^{N} log(p(o_n|\lambda)) \tag{2.8}$$

Fitting GMMs to the acoustic observations is usually done by means of the Expectation Maximization (EM) algorithm [63], which is a reliable option when an abundant amount of data is available. That is, however, not always the case when dealing with audio segments in speaker recognition and diarization. A way to overcome such a constraint is the use of a Universal Background Model (UBM) [64]. The UBM needs to be trained in a large amount of training data that comprises an equally large number of speakers to generate a large GMM (with K usually ranging between 512 and 4096 components). Its objective is to provide with a generic, speaker-independent statistical speaker template that may be used as reference for comparison with speaker-specific models adapted from the UBM itself. However, when given a set of unseen acoustic features of a limited length in a speaker recognition or diarization scenario, and a previously trained UBM model, Maximum A Posteriori (MAP) adaptation have been proven [62] more reliable to estimate speaker-dependent GMMs with regard to that of EM. Finally, speaker verification comparisons are done by log-likelihood ratios between the two possible hypothesis H_0 (the test utterance is more similar to the speaker-dependent GMM) and H_1 (the test utterance is closer to the UBM and the verification should thus be rejected).

Whilst most of the development on GMM based approaches to speaker modelling have been focused on speaker recognition, the use of GMMs has also been active in the past for SD with approaches based on Hidden Markov Models (HMM), which are used

to model the transition probabilities between speakers modelled by GMMs. Examples of such systems would be [65] or [66]. Alternatively, and following the development of the GMM-UBM paradigm towards the joint-factor analysis and i-vector scenes introduced in the next section, the concatenated means of speaker-dependent GMMs, or *supervectors* were also employed for speaker recognition [67, 68, 69] and diarization [70].

i-vectors

i-vectors have provided state-of-the-art performance in applications related to speaker modelling until very recently. These were proposed as a development of the joint-factor analysis (JFA) [71, 72] approaches that followed the development of *supervectors*. The premise upon which the theory of *supervectors* builds is that, given a set of speaker-dependent GMM models with K components fitted to acoustic features of dimension D, a compact, fixed-length and discriminative representation can be derived from the concatenation of the mean vectors and (diagonal) covariance matrices of each Gaussian component. JFA was proposed as a mean to decompose the high-dimensional space in which *supervectors* live into smaller-dimensional subspaces. Following this argument, an acoustic sample in the form of a sequence of acoustic features can be represented by a *supervector* M, which is divisible as a linear combination of speaker s and channel c dependent *supervectors* so that:

$$M = s + c \tag{2.9}$$

These two subspaces may be compressed to a compact form representable through the following two equations:

$$s = m + V \cdot y + F \cdot z \tag{2.10}$$

$$c = U \cdot x \tag{2.11}$$

in which m represents a speaker-independent *supervector* similar to that derivable from the concatenation of the UBM. V and U are two low-rank matrices that comprise the, respectively, speaker and channel subspaces, and, finally, a $KD \times KD$ matrix F which plays a similar role to that of MAP adaptation in the GMM-UBM paradigm by capturing the residual speaker variabilities not represented by V.

The concept of the i-vectors [73, 74] was proposed in an attempt to incorporate the speaker and channel variations into a single *total variability* low-dimensional subspace, providing a simpler, yet powerful discriminative analysis of the acoustic space. Similarly to the JFA proposition, the i-vector paradigm formulates the space in which a *supervector*M

lives as:

$$M = m + T \cdot w \tag{2.12}$$

where m is once again the speaker-independent *supervector*, and T is the low-rank *total variability* matrix (comprising the variability in the V, F and U subspaces). T is usually learned using the EM algorithm, and w is a low-dimensional random vector with a Gaussian, normal distributed prior $\mathcal{N}(0, I)$, whose components are referred to as *total factors*. i-vectors, given a test utterance $\mathcal{X}$ are thus derived as the posterior expectations of w. A detailed discussion about i-vectors and their implementation may be found in [75]. The i-vector approach does then act as a front-end extractor for a given utterance, but it does not apply any explicit session compensation technique or scoring. In consequence, several techniques were developed to enhance the robustness of i-vectors, like whitening length normalization [76]. Whitening consists of a normalization of the i-vector space, so that their covariance matrix is transformed into the identity matrix. Length-normalization reduces the mismatch between training and testing i-vectors through a projection into a unit sphere. For scoring, even though metrics such as the cosine distance may be used, *probabilistic linear discriminant analysis* (PLDA) [77] is usually employed. PLDA is a powerful generative framework that performs both session compensation and score computation, which led to improved state-of-the-art performance at the time of its publication.

Speaker diarization has greatly benefited from the increased speaker discriminability of JFA [78, 79] and i-vectors. In [80] the authors proposed the unsupervised dimensionality reduction of i-vectors by means of principal component analysis (PCA) leading to significant increases in performance. The work in [81] proposed unsupervised means of calibration and the inclusion of PLDA in the pipeline, followed by work in [82] where i-vectors continued to define state-of-the-art in SD.

Another interesting line of research is that related to PLDA adaptation for the in-session content of SD datasets. Given the supervised character of PLDA training, domain mismatches between training and test sets may lead to degraded performance. In counterbalancing such limitations some lines of research have been proposed using unsupervised methodologies. Variational Bayesian (VB) methods for unsupervised PLDA adaptation were introduced in [83] and applied to an i-vector based diarization in [84]. Further developments were presented in [85, 86, 87], allowing for a significant increase in performance. An alternative line to unsupervised PLDA adaptation is that proposed initially in [88] and followed by [89], in which PLDA model parameters are adjusted in an iterative fashion following multiple passes of a SD system in a *self-trained* approach.

Deep learning: speaker embeddings

The development of DL techniques is currently boosting an increase in performance in speaker modelling for both speaker recognition and diarization, and based on the leveraging of very large amounts of training data. Whilst the i-vector and GMM-UBM paradigms could be trained unsupervisedly (albeit not the PLDA model), this is not the case for DL. Algorithms providing with state-of-the-art performances based on DL are largely based in classification tasks.

First attempts to incorporate DNNs in the speaker verification paradigm were timid and attempted to substitute and/or enhance necessary modules of the i-vector paradigm by discriminatively trained DNNs. The work in [90] introduced the use of a DNN trained in an automatic speech recognition (ASR) task to replace the standard GMMs that form a UBM to produce frame alignments and the collection of sufficient statistics, in an approach close to that of [91]. An alternative line of research [92] incorporated a time-delay neural network (TDNN) [93, 94] for similar purposes. These enhancements were tested in SD in [95] with positive results.

The fundamental leap in performance did however arrive following methodologies based on DNNs trained to discriminate *explicitly* between speakers in a training set. These neural networks trained in classification tasks gave place to the development of the concept of speaker embedding. A clear definition is that given in [6](in one of their first use in SD): speaker embedding are features taken from the hidden layer neuron activations of DNNs when those are learned as classifiers to recognize over thousands of speaker identities in a training set. Although learned through identification, speaker embeddings are shown to be effective for speaker verification. Additionally, they are capable of characterising speakers unseen in the training set. While the work in [6] was based on a feed-forward DNN, multiple other network topologies have been used satisfactorily. The work in [96] used LSTMs to generate *d-vectors*, which were used to report state-of-the-art performance in SD in [8]. Authors in [57] used LSTMs too, but trained by means of the triplet-loss paradigm, a similar approach of which was applied to SD in [97]. [98] incorporated the aforementioned TDNNs in combination with a statistical pooling layer, in an approach that derived in the so-called *x-vectors* [5]. X-vectors currently constitute the most recent version of *reproducible* (as per that of [96] trained on non-public data) state-of-the-art in text-independent speaker verification. X-vectors were also successfully tested in SD in [7] and [99].

2.5 Clustering

At this stage of the SD pipeline speaker-discriminative representations are readily available to be clustered into a SD hypothesis. The right functioning of the clustering algorithm is critical for the good performance of the system, and many different methods have been proposed in the literature. All these algorithms try to deal with the highly unstructured presentation of the data in SD, and the commonly added difficulty of not knowing the number of speakers present in a session.

Hierarchical clustering

Hierarchical methods to clustering rely on some sort of initialisation of speaker clusters, i.e. random initialisation, segment-level initialisation, or speaker segmentation based initialisation, and operate upon iterative, nested operations of merging and separation of clusters. A thorough analysis of the differences in hierarchical approaches is given in [100], and its main to variants are introduced as follows:

Divisive hierarchical clustering: A first, less common approach is that of divisive hierarchical clustering (DHC) which develops a top-down, general-to-specific approach to speaker clustering. A single (or alternatively a small number) of clusters is used as seed for the clustering process, which is iteratively split into smaller speaker clusters until a stopping criterion is met and the diarization output is fixed. Examples of such systems are [101, 102, 103].

Agglomerative hierarchical clustering: Bottom-up agglomerative hierarchical clustering (AHC) approaches are more commonly used nowadays in SD systems. They operate in an opposite manner to that of DHC, and apply a specific-to-general methodology which allows initial clusters to be *ideally* purer from the initial iteration. A cluster-to-cluster similarity matrix is computed at each pass of the AHC algorithm to decide which clusters to merge before continuing with the iterative process. This approach to clustering is straight-forward, and the merging stops when similarity between clusters falls within an empirically optimised threshold, determining the final number of speakers in a session. Applications of AHC to SD are thus abundant, and continue to provide with state-of-the-art performance in some acoustic domains [99]. Whilst easy to put in practice, it may be argued that AHC incurs in what is a greedy sort of decision-making by not necessarily allowing for re-assignment of segment-to-cluster relationship at every iteration. This may be solved though by means of a simple segment-to-cluster re-assignment operation at the beginning of every clustering step. Such an approach is used in [104] and [105].

A slightly different approach to AHC worth mentioning is that applied to SD via the information bottleneck (IB) approach [106, 107] in [108], which operates upon acoustic features directly by using a non-parametric framework derived from the Rate-Distortion theory [109].

Bayesian analysis

Another branch of research worth noting here is that which explores Bayesian analysis for the task of SD. A first application of variational Bayes to SD was proposed in [110], and further developed in [111, 112] by leveraging the use of eigenvoice priors for VB inference. In parallel, non-parametric Bayesian diarization solutions were also being proposed by combining hierarchical Dirichlet processes (HDP) [113] with HMMs [114], and applied to SD in [115]. In combination of these two lines of work, the authors in [116] recently reported an enhanced version of that in [111] in that it incorporates the HMMs of [115] to model speaker transitions. The resulting work has achieved state-of-the-art performance in SD performance in various domains [116, 117].

Other approaches

A variety of other clustering methods have been proposed in the literature. In [118] integer linear programming (ILP) proposes an approach that replaces AHC by a global clustering process. ILP attempts to minimize the number of clusters and their dispersion by means of optimising a single objective function. K-means has been proven reliable in [80] in contexts where the number of speakers is known, albeit these are not always of interest. Spectral clustering (SC) has also been applied [8, 119, 120], motivating part of the presented in Chapter 6. Mean-shift [121] based methodologies were also applied upon SD solutions as mean to clustering in [122] following their development in [123, 124].

All of the clustering methods exposed in this section are offline clustering algorithms. These operate upon an entire, finalised set of data observations. Offline clustering is, however, simpler than its online counterpart. This is due to offline methods being capable of linking temporally separated speaker representations from the first iteration of a hierarchical algorithm, which online methods cannot replicate. While a fully-fledged offline diarization could potentially be operated at time t with every new observation as proposed in [15], applications in need of online diarization do also tend to require the low-latencies associated to it. Under such constraint, the full recomputation of offline clustering approaches may be too computationally expensive. However, interesting methodologies have been proposed in the literature that perform fully online SD [16, 17, 125]. When the number of speakers in the conversations of a dataset are known the problem becomes

relatively easier [126], and developments have been proposed in the form of online adaptive modelling [127], or the use of semi-supervised approaches that leverage some sort of cluster initialisation [128]. State-of-the-art performance in SD using online clustering has, however, recently been achieved in controlled domain constraints [8], which provides scope for future research and growth for this kind of clustering algorithms.

2.6 Resegmentation

A last, optional module in the pipeline of SD is that refining the boundaries generated by the clustering algorithm and/or including short segments which may have been removed for more robust clustering performance [129]. A traditional approach to resegmentation is that of an ergodic HMM in which speakers modelled by means of GMMs are used as HMM state distributions. Viterbi alignment is then computed to obtain the final result. Alternatives to resegmentation have however gained importance in recent years thanks to enhancements proposed in the literature. The VB inference methodologies such as that of [116] introduced above, have been extensively used as a resegmentation method following a first clustering solution derived from i-vectors/speaker embeddings, yielding positive increases in performance [7, 99, 130]. Neural networks have similarly allowed for refined boundaries definition. In [131], an initial IB diarization system is applied only to generate speaker pseudo-labels which may be used to train an artificial neural network (ANN) capable of enhancing the speaker-discriminative capacity of acoustic features. A similar approach is that successfully used in [132] by means of LSTM networks.

2.7 Evaluation and metrics

2.7.1 Speaker diarization

The most common metric in evaluating diarization performance is the diarization error rate (DER). This scoring method, originally proposed in the context of the National Institute of Standards and Technology (NIST) Rich Transcription (RT) evaluations [133], is thus used in the experimental results reported in this thesis.

The DER considers errors derived from VAD, segmentation and clustering stages of the SD pipeline, and is defined as:

$$DER = E_{spk} + E_{FA} + E_{miss} + E_{OV} \tag{2.13}$$

where E_{spk} corresponds to the speaker error rate, or percentage of speech time

assigned to an incorrect speaker, E_{FA} relates to the error generated by the VAD system by labelling non-speech content as speech. E_{miss} represents the opposite mistake by the VAD system: speech time being wrongfully discarded as non-speech. Finally E_{OV} is the percentage of error motivated by the insufficient annotation of overlapping speech. Given rich reference annotations in which two or more speakers interact at the same time, a SD system should ideally be able to label all the involved speaker identities.

In order to account for possible imprecisions in the human annotation of the diarization references, a collar of forgiveness is usually applied. In consequence, diarization hypothesis boundaries within the 0.25s surrounding the boundaries of the diarization references are considered as correct. Unless stated otherwise (such as in the results reported in Chapter 6), this criterion is also applied throughout this thesis.

2.7.2 Speaker recognition

In the assessment of automatic speaker verification (ASV) two different kind of errors are mainly considered, often used in biometric authentication applications. They can be defined as [4]:

- False accept/alarm (FA): the erroneous acceptance of an impostor speaker with regard to a, different, claimed identity.
- False reject/missed detection (FR/MD): the rejection of a legitimate speaker with regard to to its previously enrolled speech.

These are often referred to as rates (FAR and FRR/MDR) when divided by the total amount of impostor/legitimate attempts in a test set. Given a test utterance and a claimed identity which constituting a trial, an ASV system generates a single scalar score. In a practical scenario trials scores exceeding a certain threshold θ will be accepted and the rest will be rejected. The optimisation of the threshold θ does thus have implications in the amount of FA or MD present in a list of trials, of importance with regard to the final application of the ASV systems. An exceedingly low threshold results in a high number of FA, whilst the contrary results in a high number of MD. In order to provide with an application-independent, generic interpretation of the performance of an ASV system, the equal error rate (EER) is often used. The EER is defined as the working point in which the threshold θ generates a FAR = FRR/MDR. The EER is thus used in some speaker recognition experimental results presented in this thesis such as those of Chapter 4 in which the final application is not considered. In addressing the relationship between ASV performance and final application needs, the NIST speaker recognition evaluations (SRE) [134] use a detection cost function (DCF) as primary

metric. The DCF introduces associated costs/penalties to the FAR and MDR/FRR, C_{miss} and $C_{falsealarm}$ [134], and provides a priori target speaker probabilities. The DCF or C_{det} is thus defined:

$$C_{det}(\theta) = C_{\text{miss}} \times P_{\text{target}} \times MDR(\theta) + C_{\text{fa}} \times (1 - P_{\text{target}}) \times FAR(\theta) \tag{2.14}$$

In consequence, in the context of our work related to security-based applications the C_{det} is also considered in Chapters 8 and 9.

2.8 Summary

This chapter provides with a brief literature review of the traditional and more modern, state-of-the-art methods employed in the different modules that compose a SD pipeline.

Part I

Domain robust and efficient speaker diarization

Part I of this thesis focuses on BK modelling and the quest for efficient, domain-robust solutions to speaker diarization (SD). Contributions relate to the enhancement of a baseline system through modifications to front-end processing, speaker change detection, clustering and system combination. Chapter 3 reviews the BK speaker modelling technique. Chapter 4 presents the first application of multi-resolution spectral analysis-based feature extraction to BK speaker modelling. It exploits the constant Q transform. Chapter 5 introduces the use of BK modelling for speaker change detection (SCD). Contributions include a novel method to binary key background model (KBM) composition. Chapter 6 puts the focus on clustering and reports the first application of spectral clustering (SC) to BK-based SD. Chapter 7 reports efforts to combine BK-based and deep learning based solutions to SD. Combination is achieved through two different fusion approaches; either using a similarity matrix, at the system level, or by hypothesis-level fusion.

Chapter 3

Binary key speaker modelling: a review

This chapter provides a review of the binary key (BK) speaker modelling technique. The material puts into context the developments proposed in this part I of the thesis which are outlined in Chapter 1. The presentation is organised as follows. The introduction presented in Section 3.1 highlights the motivations of the original BK method and its principal characteristics. Section 3.2 introduces the binary key background model (KBM) which plays a fundamental role in the application of the BK technique to both speaker recognition and diarization. Section 3.3 describes the process by which acoustic features are transformed into the binary domain. Section 3.4 describes the process to obtain segment level representations based upon BK speaker modelling. These segmental, speaker-discriminative representations need appropriate metrics for their comparison and application that exploit their computational simplicity in the binary domain. These are introduced in Section 3.5. Section 3.6 offers a brief review of the advancements and modifications brought by other authors to the original BK method in speaker recognition, diarization, and other related applications. Section 3.7 presents the BK speaker diarization (SD)system that serves as a baseline for the rest of work presented in part I of the thesis. A summary is presented in Section 3.8.

3.1 Introduction

The traditional approaches to state-of-the-art voice biometrics rely on complex statistical analysis and large amounts of background data to learn from. Motivated by the need to compensate for the numerous sources of variability present in audio channels and

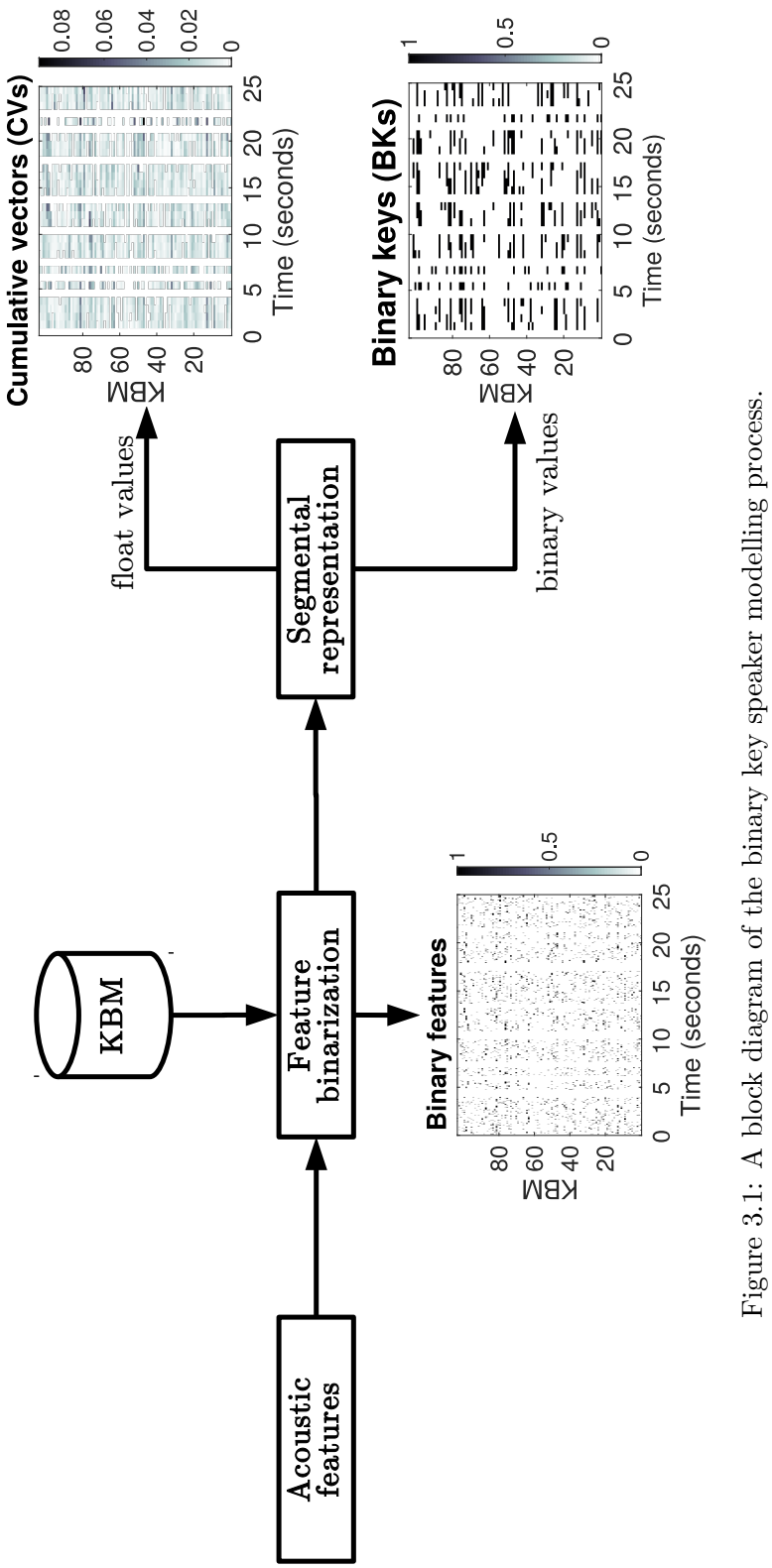


Figure 3.1: A block diagram of the binary key speaker modelling process.

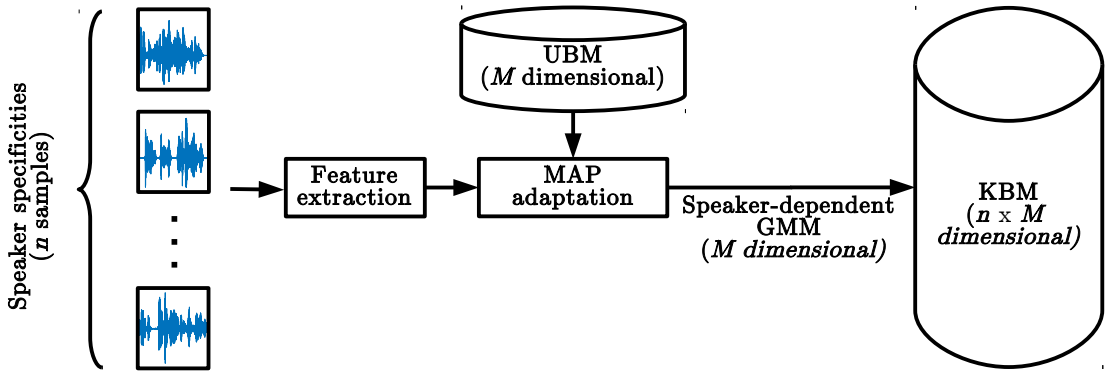


Figure 3.2: KBM composition method for speaker recognition tasks. Speaker specificities act as anchor models to bridge the speaker-independent space modelled in the UBM to a speaker-dependent equivalent in the KBM.

recordings, these algorithms have evolved in performance but also in requirements. Typically, they demand high computational capacity in addition to specific hardware, i.e. GPUs [135], needed to process massive amounts of labeled data from which deep neural networks learned via supervised learning. Even with huge computational resources, the adaptation of such algorithms to perform reliably in unseen domains remains difficult and time-consuming.The difficulty arises from the need to collect and label sufficient data that is representative of the new domains. Both the labelling of data and the adaptation of existing models are costly and time-consuming. Data scarcity characterises many practical applications of voice biometrics, e.g. those including languages from developing countries [136]. For voice based human-machine interaction to be used at a global scale, techniques that are easily adapted to new scenarios or can dispense with training data can offer an advantage.

BK speaker modelling was introduced in [137] as an alternative to traditional statistical modelling techniques in the context of voice biometrics. Originally proposed for the task of speaker recognition [9], BK speaker modelling was proposed as a means of addressing these limitations. It offered an approach to discriminate between speakers that, whilst being computationally inexpensive, reduces the demand for background training data for some applications, meaning it can be readily and easily adapted to new domains. An overview of the BK speaker modelling approach is illustrated in Figure 3.1. First, (left of Fig. 3.1), acoustic features are projected to a binary space. This transformation is performed using a binary Key Background Model (KBM). Binary features are then aggregated to generate speaker-discriminative segmental representations in the form of Cumulative Vectors (CVs) composed of float values, or Binary Keys (BKs) in the form of binary values. The different steps in this process are described in detail in the following, along side variations in processing for speaker recognition and SD.

3.2 Binary key background model

The transformation of acoustic features into a speaker-dependent binary space is dependent on the binary Key Background Model (KBM). The KBM plays a role similar to that of the UBM in more traditional approaches to ASV (see Chapter 2). The UBM is a GMM representation of the average speaker acoustic space and is learned from large quantities of training data. This space is typically learned in an unsupervised manner using the EM algorithm. Each Gaussian component λ_i of the GMM of dimension M models a region of the acoustic space defined by a mean vector μ_i, a covariances matrix Σ_i, and a weighting factor p_i. Similarly, a KBM of dimension N is defined by Gaussian components λ_i that represent the acoustic space not so much in an average sense, but more in a speaker-discriminant sense. The following sections describes approaches to KBM learning. Section 3.2.1 focuses on speaker recognition whereas Section 3.2.2 has a focus on SD.

3.2.1 Speaker recognition

The original work in [137] introduced a procedure for KBM learning that is based upon the concept of anchor models [138, 139]. The procedure is illustrated in Figure 3.2. It operates upon a GMM-UBM of dimension M, trained using background data. A number n of speech samples collected from different speakers, defined as *speaker specificities* (left of Fig. 3.2), are used to learn a set of speaker-dependent GMMs using MAP adaptation of the UBM (center of Fig. 3.2). The KBM is composed by the concatenation of these n speaker-dependent GMMs, and is thus of dimension $N = n \times M$ (right of Fig. 3.2). The selection of the *speaker specificities* that generate the final KBM is important to the resulting ASV performance, with [9] emphasizing the importance of gender balance among the chosen *specificities*.

3.2.2 Speaker diarization

The procedure for learning the KBM for the task of SD is fundamentally different to that adopted for ASV. In contrast to most approaches to SD which employ *external* training data, the work in [140] proposed a procedure that exploits the KBM without needing any. Instead, the KBM is learned using speech present in the test session itself. The process is illustrated in Figure 3.3. First, the acoustic space of the test session is sampled with sliding and overlapping windows (top of Fig. 3.3). A multivariate Gaussian distribution $\lambda_i \sim \mathcal{N}(\mu_i, \Sigma_i)$ is then fitted to the acoustic features f_{λ_i} present in the i^{th} step of the sliding window, thereby producing a pool of G in-domain Gaussian distributions, where the dimension G is related to the length of the audio session for a given window shift. The

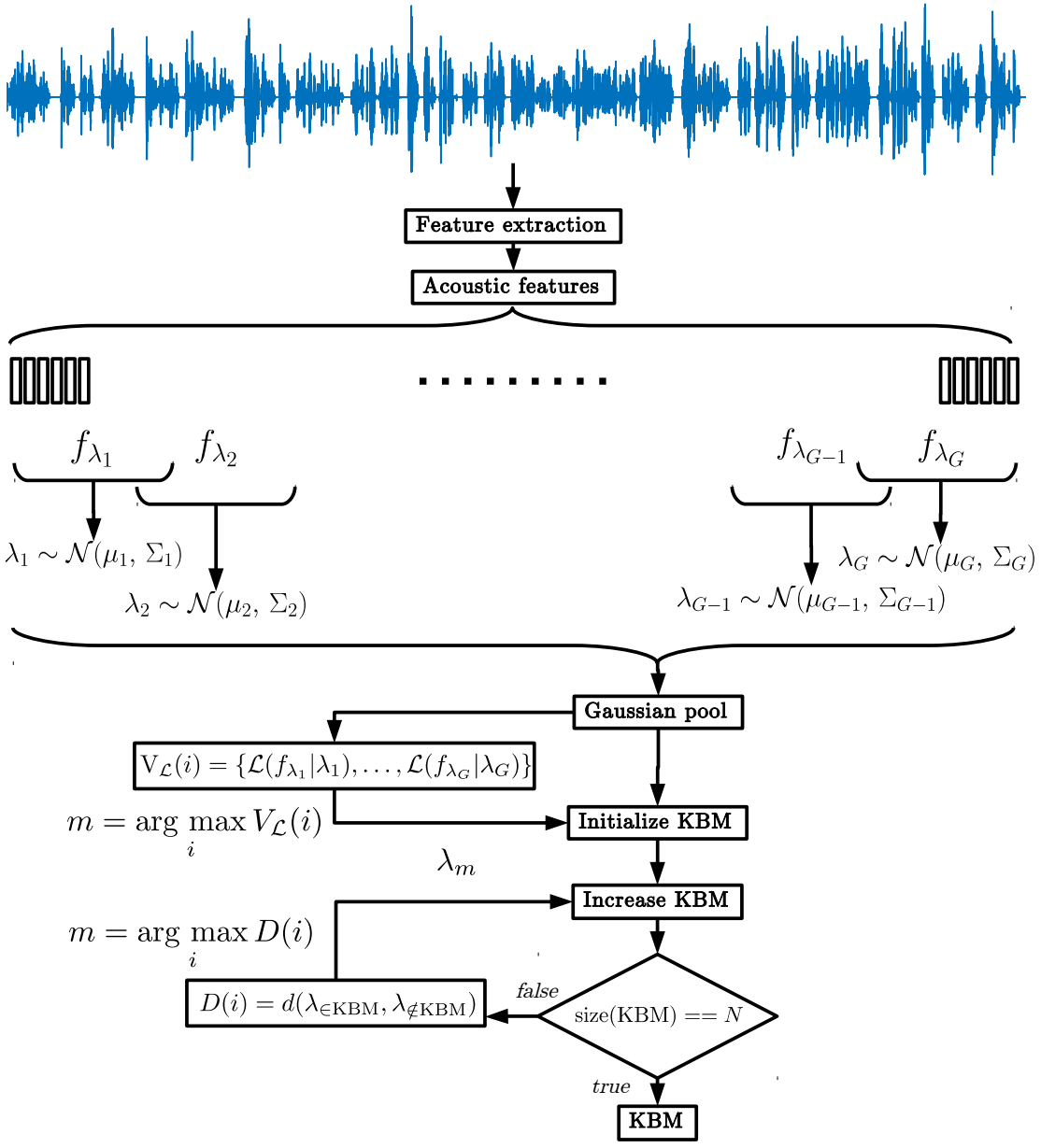


Figure 3.3: KBM composition method for SD tasks. Feature vectors within a sliding window f_{λ_i} are fitted to a Gaussian λ_i. The set of Gaussians over all windows are added to the pool. The best-fitting Gaussian, judged by the vector of likelihoods $V_{\mathcal{L}}(i)$, is selected to initialize the KBM. An iterative process in which the distance between the Gaussian elements in the KBM $\lambda_{\in KBM}$ and the remaining elements $\lambda_{\notin KBM}$ is computed so that the most dissimilar Gaussian element not already in the KBM is added until the desired KBM size is reached.

likelihood of the features $\mathcal{L}(f_{\lambda_i})$ given the corresponding model, $\mathcal{L}_{\lambda_i}$, is then calculated according to:

$$\mathcal{L}_{\lambda_i} = \mathcal{L}(f_{\lambda_i}|\lambda_i) = \mathcal{L}(f_{\lambda_i}|\mathcal{N}(\mu_i, \Sigma_i)), \tag{3.1}$$

in order to determine a vector of likelihoods $V_{\mathcal{L}}(i) = \{\mathcal{L}(f_{\lambda_1}|\lambda_1), \mathcal{L}(f_{\lambda_2}|\lambda_2), \ldots, \mathcal{L}(f_{\lambda_G}|\lambda_G)\}$.

Next, a subset of $N < G$ Gaussian components from the Gaussian pool are selected to compose the KBM. The selection algorithm is based upon a *minimum redundancy, maximum relevance criterion* [141]. The selection process starts (center of Fig. 3.3) by identifying the Gaussian component λ_m with the highest likelihood in $V_{\mathcal{L}}(i)$ according to:

$$m = \arg\max_i V_{\mathcal{L}}(i). \tag{3.2}$$

An iterative process is then applied to complete the KBM (bottom of Fig. 3.3). Each iteration begins with the computation of the distance between the components of the pool that belong to the KBM $\lambda_{\in \mathrm{KBM}}$ and components remaining in the Gaussian pool, $\lambda_{\notin \mathrm{KBM}}$. The distance measure between distributions proposed in [140] is the Symmetric Kullback-Leibler Divergence (KL2)(Chapter 2), defined as:

$$D_{\mathrm{KL2}}(P \parallel Q) = D_{\mathrm{KL}}(P \parallel Q) + D_{\mathrm{KL}}(Q \parallel P), \tag{3.3}$$

which offers a symmetrized alternative over the standard KL Divergence (KL). Given two distributions P and Q, the KL divergence is defined as:

$$D_{\mathrm{KL}}(P \parallel Q) = \frac{1}{2}\left(\mathrm{tr}\left(\Sigma_Q^{-1}\Sigma_P\right) + (\mu_Q - \mu_P)^{\mathsf{T}}\Sigma_Q^{-1}(\mu_Q - \mu_P) - k + \ln\left(\frac{\det\Sigma_Q}{\det\Sigma_P}\right)\right) \tag{3.4}$$

where Σ_P, μ_P, Σ_Q and μ_Q are, respectively, the covariance matrices and mean vectors of the distributions P and Q, and where k is the dimension of the data. To conclude the iteration, the Gaussian component with the highest distance D_{KL}, λ_m, is added to the KBM. Similarly to that of Equation 3.5, here m is simply determined as:

$$m = \arg\max_i D_{KL2}. \tag{3.5}$$

The KBM is complete when N elements of the Gaussian pool have been selected. Naturally, the KBM dimension N should be session-dependent. N should be large enough to cover all the variability in the Gaussian pool, but also small enough so as to avoid selection of redundant Gaussian components . In practice, the parameter N is empirically optimised. As is common in the literature, and as is also the case of this thesis, results are reported as a function of N.

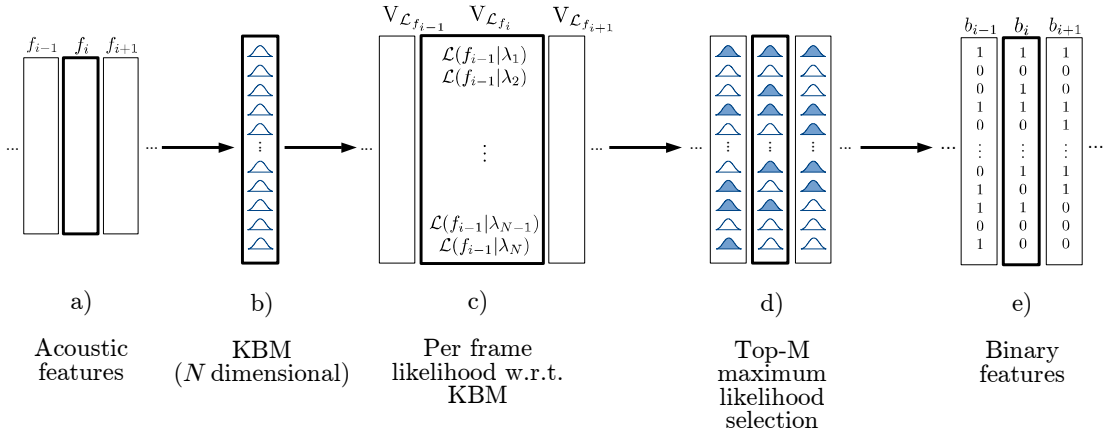


Figure 3.4: Process of feature binarization. Acoustic features are mapped to the Gaussian elements in a KBM in a per frame basis by calculating their likelihoods. Top M Gaussian elements per feature are activated and turned to 1 in the final binary feature representation.

3.3 Feature binarization

As represented in Figure 3.4, once a KBM is trained it is possible to perform the conversion of the acoustic features to the newly represented speaker-dependent domain. In order to do so, let all the feature frames present in a piece of audio be $F_T = \{f_1, f_2, \ldots, f_t\}$. As introduced in [137], the likelihood of an acoustic frame $f_i \in F_T$ belonging to the N Gaussian elements of the KBM is calculated. For a Gaussian element of the KBM λ_n, a single likelihood would be

$$\mathcal{L}_{f_i} = \mathcal{L}(f_i|\lambda_n) = \mathcal{L}(f_i|\mathcal{N}(\mu_n, \Sigma_n)). \tag{3.6}$$

For the complete set of KBM Gaussians a vector of likelihoods per acoustic frame is obtained so that

$$V_{\mathcal{L}_{f_i}} = \{\mathcal{L}(f_i|\lambda_1), \mathcal{L}(f_i|\lambda_2), \ldots, \mathcal{L}(f_i|\lambda_N)\}. \tag{3.7}$$

In order to obtain a binary representation b_i of the feature f_i, the vector of likelihoods $V_{\mathcal{L}_{f_i}}$ is quantized. The top-M elements with the highest likelihoods in the vector $V_{\mathcal{L}_{f_i}}$ are *activated* and set to 1, whereas all the other elements in binary vector are set to 0. As explored in [9], the choice of M has an impact on performance: a too restrictive number of activations will lead to highlighting Gaussian elements common to all of the features, and erroneously included in the KBM. On the other hand, a number of activations that is too high will eventually lead to speaker-discriminative differences being lost, effectively degrading the performance of a binary key-based system.

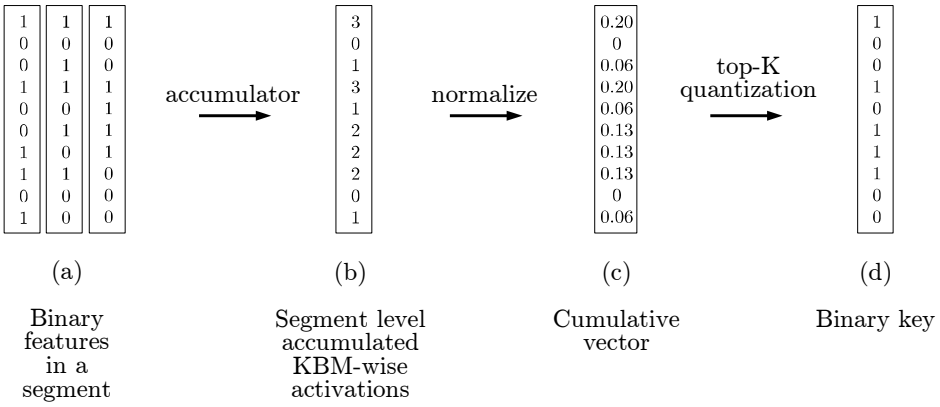


Figure 3.5: Extraction of segmental level representations.

3.4 Segmental representations

Speaker recognition systems generally need a certain amount of speech data so that they operate upon sufficiently discriminative segment level representations. Typically, at least a few seconds. Binary key speaker modelling is not an exception to this requirement. Binary features also need to be processed in some manner to provide robust segmental representations. In order to generate these representations, accumulative and further quantizing steps need to be applied to produce, respectively, Cumulative Vectors (CVs) and Binary Keys (BKs). This process is depicted in Figure 3.5.

A CV is defined as a vector of floats with the same dimensionality as the KBM (and consequently the dimension of the features represented in the binary domain). To obtain a single CV, and following the process illustrated in Figure 3.5, binary features (a) are accumulated element-wise at the utterance level (b). In order to provide length-independent representations, the resulting vectors are also normalized over the total sum of activations in a utterance, thereby giving the CV (c). CVs represent a vector of weights that relate the relationship of the Gaussian elements in the KBM to the speech content in a utterance.

Segmental level binary representations are obtained as a further quantization step applied to CVs. In the process transforming the acoustic features to the binary domain (Section 3.3), the top-M elements corresponding to the highest likelihood with regard to the elements in the KBM are set to 1. Similarly, here, the top-K elements with the most activations in a CV are set to 1 to generate the final BKs.

3.5 Similarity metrics for binary key speaker modelling

With BK speaker modelling having been designed with computational efficiency in mind, metrics that allow for fast computation and comparison of CVs and BKs are necessary. The work in [105] proposed the use of the cosine similarity as a means of comparing CVs. Given two CVs to compare, CV_a and CV_b, it is defined as:

$$S(CV_a, CV_b) = \frac{CV_a \cdot CV_b}{\|CV_a\| \, \|CV_b\|} \tag{3.8}$$

The cosine similarity focuses on the angular difference between two CVs. Values in the cosine distance for CVs, given that CVs values are limited between 0 and 1 after their length normalization, are also bounded between 0 and 1. CVs that are aligned in the same direction, will score a high cosine similarity close to 1. Nearly orthogonal pairs would result in a cosine similarity close to 0.

For BKs, different metrics have been employed for different applications. Their pure binary format allows a number of distance measures to be borrowed from information theory applications. For speaker recognition, the length-normalized Hamming weight of the logic AND comparison can be applied [9]. In that case, the similarity between two BKs, BK_a and BK_b, can be defined as

$$S(BK_a, BK_b) = \frac{1}{N} \sum_{i=1}^{N} (BK_a[i] \wedge BK_b[i]) \tag{3.9}$$

where $\wedge$ specifies the bit-wise AND operator and N is the BK dimension.

For the task of SD, the work in [10] proposes the use of the Jaccard similarity, defined as

$$S(BK_a, BK_b) = \frac{\sum_{i=1}^{N} (BK_a[i] \wedge BK_b[i])}{\sum_{i=1}^{N} (BK_a[i] \vee BK_b[i])} \tag{3.10}$$

where $\vee$ is now the bit-wise OR. This operator serves as an additional measure to *diss*imilarity between pairs of BKs, by exploiting the in-session training of the KBM and the locality of the representations.

3.6 Recent improvements and use cases

Since its introduction, improvements in BK speaker modelling have been explored not only in the speaker recognition and diarization tasks, but in a number of other related applications.

3.6.1 Recent improvements for speaker recognition

A line of developments made in its usage for speaker recognition have explored enhancements in the process of composing the KBM. In [141], the KBM is composed in a two-stage process that merges the methods for KBM composition explored in speaker recognition [137] and SD [140]. In a first stage, a primary KBM is composed following the anchor model methodology described in Section 3.2.1, by replicating the UBM into several MAP-adapted *speaker specificities*. Second, the definitive KBM is obtained by applying the same *minimum redundancy, maximum relevance*-based algorithm described in Section 3.2.2. This step aims to selecting only the most discriminative specificities. Other attempts to improve the discriminability in the KBM composition explored temporal [142] and neighbooring [143] relationships among specificities.

In a different direction, the work in [141] reported the use of session compensation techniques such as nuisance attribute projection (NAP) [68, 69]. Originally applied to GMM-UBM derived supervectors, NAP is a kernel independent technique that tries to minimize channel variability effects, and can thus be applied to BK speaker modelling in the context of speaker recognition, be it in the form of CVs [144] or BKs [141].

Finally, metrics related to the comparison of BKs were proposed, including the successful application of PLDA (see Chapter 2) to BK scoring [142], or BK-specific metrics, such as Intersection and Symmetric Difference (ISDS) [145].

3.6.2 Recent improvements for speaker diarization

Following its initial implementation for SD [140], enhancements were reported, mainly by Delgado [105]. This work focused on reducing execution time in exchange for a small decrease in performance. These enhancements were obtained not only by improvements to the process of BK/CV extraction, but also to the optimization of different modules in the SD pipeline.

For the process of KBM composition for SD, [146] explored the use of alternative metrics to the KL2 divergence to discriminate among the Gaussian components in the Gaussian pool. Despite the KL2 divergence having been used largely for voice biometrics, the different operations necessary to compute it, which include traces, determinants and inversions, make it computationally expensive. The authors explored the use of the cosine distance, defined between two Gaussian components λ_a and λ_b as:

$$D_{cos}(\lambda_a, \lambda_b) = 1 - S_{cos}(\mu_a, \mu_b), \tag{3.11}$$

where $S_{cos}(a, b)$ is the cosine similarity, defined as in Equation 3.8, and μ_a and μ_b are

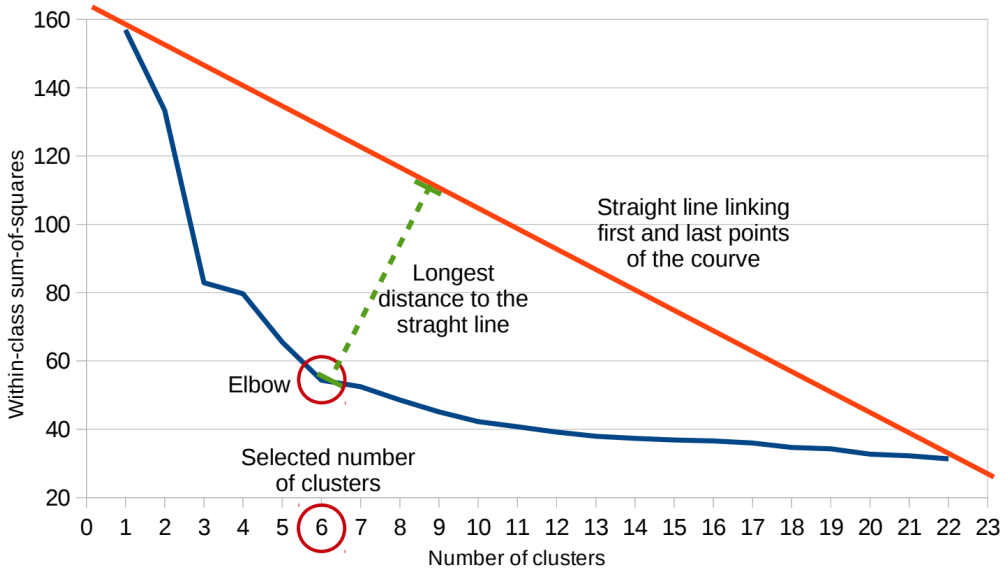


Figure 3.6: An example of the elbow criterion for number of cluster estimation. applied over the curve of within-class sum-of-squares per number of clusters. The point with longest distance to the straight line is considered the elbow.

the mean vectors representing the Gaussian components. This solution proved to offer a lightweight yet similarly effective alternative to the task of KBM composition.

Improvements were also explored for the problem of the stopping criterion in an AHC process, of particular relevance to the SD problem. The first SD system based on binary keys [140] explored a T-test T_s-based metric [147]. Integer Linear Programming (ILP) [118], was also tested as a possible alternative [148]. Despite the promising results reported in both [140] and [148], these also showed the significant margin for improvement in the system given an appropriate AHC stopping criterion. Later work by the same authors [149] continued this exploration using an elbow criterion, illustrated in Figure 3.6. This mechanism is based on the value of the Within-Cluster Sum of Squares (WCSS) of a set of given clusters, which is usually employed as an objective function in other clustering algorithms such as k-means. The method proposed in [149] is based judging the evolution of the WCSS value as the iterative steps of a bottom-up AHC process take place (dark blue line in Figure 3.6, bottom-up process goes from right to left). During the AHC, meaningful merges, i.e. merges of same-speaker clusters, are expected in the first steps of a bottom-up approach (bottom right WCSS values on Fig. 3.6). This is due to initial cluster partitions being presumably pure in terms of speaker content, generating a small WCSS. Only when the AHC process starts merging clusters whose respective content is more sparsely distributed in the speaker space (possibly meaning different speakers are present in a single cluster), the WCSS value starts increasing (left values of the WCSS in Fig. 3.6). The elbow criterion proposed in [149] offers a trade-off between

intra-cluster and inter-clusters distributions based on detecting the abrupt increase of the WCSS value in the AHC. When the AHC process is finished, the chosen number of cluster (circled in red in Fig. 3.6) is selected as the point in the WCSS curve with the longest distance to the straight line linking the initial and final values of the WCSS curve(red solid line in Fig. 3.6).

Similar to the focus on session compensation introduced in Section 3.6.1, the work reported in [146] explored the application of NAP for the task of SD using CVs. NAP relies on the availability of a suitable labeled dataset to learn the necessary transformations for Intra-Session and Intra-Speaker Variability (ISISV) compensation to be effective. Different to the use of NAP for BK-based speaker recognition, in which the KBM is trained using external training data, and in which NAP can be used in a more standard manner, BK-based SD produces utterance level CVs or BKs related to the in-session trained KBM, making between-session comparisons (and thus compensation) unfeasible. In order to overcome this limitation, NAP transformations are learned and applied on a session basis. To do so, a two-stage process starts from an initial uncompensated diarization pass. The generated diarization hypothesis is then used to learn the appropriate in-session transformations that attempt to minimize ISISV.

3.6.3 Other applications

As evidence of its versatility and capacity for acoustic modelling, BKs have been used in other tasks related to speech recognition. The work in [150] explored the use of BK modelling for the task of voice activity detection in an attempt to integrate it in a BK SD pipeline. The work in [151, 152] reported the BK use to model and identify emotions in audio streams. Finally, the BK modeling technique has recently been applied in the context of template protection and cryptography for privacy preservation in the context of cancelable biometric systems [153]. In [154], BK modelling is used for the task of cohort selection for speaker recognition in encrypted domains, motivated by the heavy computational times of other techniques in such scenarios.

3.7 Baseline system for speaker diarization

Being mainly motivated by the search for improvements to training-independent SD technologies this thesis explores improvements in a number of elements of the binary key-based SD paradigm by focusing upon different elements in its pipeline. In doing so, the solution reported by Delgado [105] and the enhancements explored there, briefly described in Section 3.6.2, were used as a baseline. Following the notation and modules introduced in Chapter 1 and Chapter 2, the SD baseline represented in Figure 3.7 can

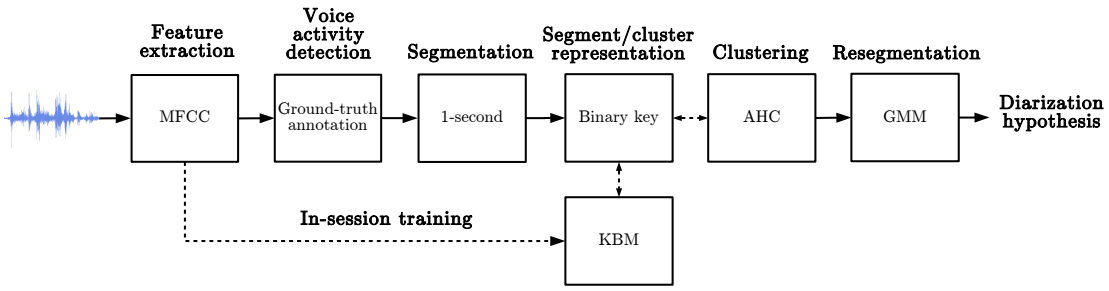


Figure 3.7: Baseline SD pipeline as considered in this manuscript.

be described as follows:

Feature extraction: Standard MFCCs are extracted over a sliding window of 25ms with a 10ms shift. 19 coefficients are employed without energy coefficient or derivatives.

Voice activity detection: No particular enhancement was pursued in this regard despite the primary results on BK-based SAD [150]. Unless stated otherwise, during the course of this thesis the SAD module relies either upon oracle ground-truth annotations or employs an externally generated SAD hypothesis[1]

Segmentation: Speech content is split into chunks of 3s with a shift of 1s, hence with an overlap of 2s.

Segment/cluster representation: In order to obtain speaker discriminative segment level representations a number of steps are applied as described in the previous sections (Section 3.2.2, Section 3.3 and Section 3.4) of this chapter. Namely:

> **Gaussian pool generation:** As described in Section 3.2.2, a Gaussian pool of dimension G is generated by fitting single-dimensional multivariate normal distributions to the content of a sliding window. The window spans 2s of speech, with an adaptive shift between windows that ensures a minimum amount of Gaussians. This is done to provide an adequate representation of the acoustic space in files of limited length. Alternatively, for files of sufficient speech content, the shift between sliding windows is limited to 0.5s to provide sufficient resolution of the complete acoustic space.
>
> **KBM training:** The KBM of dimension N is composed following the procedure described in Section 3.2.2. In contrast to the original implementation, it follows the method proposed in [146] where cosine similarity is used to discriminate between

[1]Different external SAD systems were used depending on the experiment. Details of the used configurations are given at the respective experiment setups.

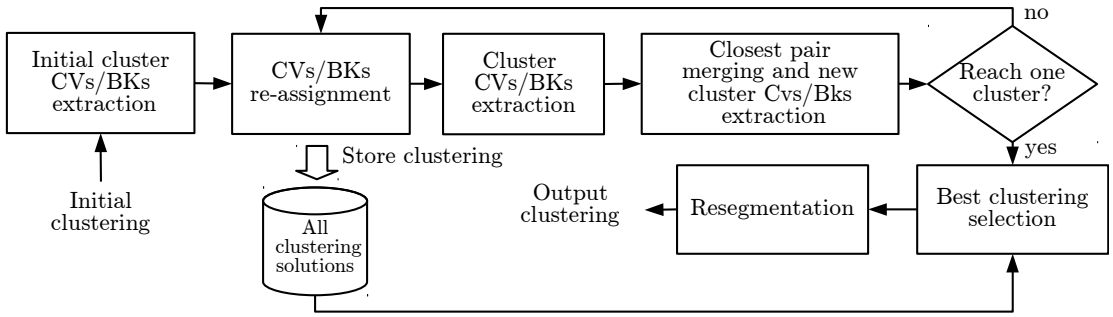


Figure 3.8: Bottom-up AHC algorithm used as baseline across this thesis.

Gaussian elements. Whereas all previous work determined N as a fixed value optimized over an entire dataset, in this thesis it was alternatively chosen to according to the length of the speech content. Defined as a percentage of the size of the Gaussian pool G, an adaptive KBM size allows for a better fitting to the variations in terms of session length that may exist within a dataset.

Feature binarization: For the process of acoustic to binary feature transformation described in Section 3.3, the top $M = 5$ Gaussians with the highest likelihoods are set to 1, while the remainder elements are set to 0. This number is selected accordingly to the results reported in [9].

CV and BK extraction: For CV extraction the segment-level binary features are accumulated element-wise and then normalized over the sum of the total activations. For BKs the top-K elements with the highest activation in the respective CVs are set to 1. This parameter K is set as percentage so that the top-$K = 20\%N$, where N is the size of the KBM, elements of a BK are 1.

Clustering:

The AHC algorithm is represented in Figure 3.8. First, to initialize the process, speech features are split into N_{init} contiguous segments of equal size. From each one of these segments an initial CV/BK is estimated to represent the initial clusters. Then, the bottom-up iterative AHC algorithm is applied to the segment and cluster CVs. At each iteration, segment-level CVs/BKs are compared to the cluster CVs/BKs by means of the cosine/Jaccard similarity and assigned to the closest cluster. Cluster CVs/BKs are then compared among themselves using the same cosine/Jaccard similarity, with the closest pair of clusters being merged and re-estimated as a single cluster, thus reducing the number of clusters by one per iteration. This process is repeated until a single cluster remains. A clustering selection algorithm is then applied. It is based upon the elbow criterion reported in [149].

Resegmentation: In order to provide a refined diarization hypothesis, a resegmentation is performed. 128-component GMMs are trained on the content of each hypothesized cluster. Then, features are compared to the models, and the resulting likelihoods are smoothed over a sliding window of 1s. Finally, each feature frame is assigned to the GMM model with the highest likelihood, thereby generating the final diarization hypothesis.

3.8 Summary

This chapter provides with a detailed review of the binary key speaker modelling technique. By representing speech segments in the form of binary representations, BK modelling is a fast and reliable alternative to other more state-of-the-art speaker modelling techniques. While relying on the use of a KBM, and being proposed initially for the task of speaker recognition, it has been for the challenging problem of SD where most of the recent advancements have been reported. This is motivated by the capacity of the KBM to be trained without the use of any external training data, a rather uncommon characteristic with respect to most of today's deep learning based solutions. Data independence makes it an inherently robust approach to SD, even in the case of within-dataset variations. By leveraging recent developments in BK-based SD, this chapter also defines a baseline SD. Despite having achieved reasonable results in the reported literature without using any external training data, baseline performance may be improved by enhancing different elements of the pipeline. These explored improvements motivate the work included in this first part of this thesis, and they are explained in detail in the following chapters.

Chapter 4

Multi-resolution spectral analysis for speaker diarization

Spectral analysis is one of the fundamental tools upon which acoustic features are built and used in speech processing tasks. Traditional approaches to spectral analysis are used to draw fixed-resolution representations of the acoustic space. While this is proven to be valid for many tasks, our particular interest in binary key (BK) speaker modelling for speaker diarization (SD) motivates exploring alternatives that represent the spectrum differently. In this chapter, acoustic features that exploit multi-resolution spectral analysis and their impact to BK speaker modelling are assessed in comparison to traditional methods. The analysis and results contributed of the EURECOM's submission to the Albayzin 2016 Speaker Diarization Evaluation [155]. Section 4.1 discusses the importance of feature extraction (FE) to BK speaker modelling. It also introduces improvements in multi-resolution spectral analysis that motivate this work. Section 4.2 gives a theoretical background to traditional spectral analysis and to the multi-resolution approaches explored. The strengths and limitations of the different spectral analysis techniques are assessed following the analysis reported in Section 4.3. Section 4.4 describes the Albayzin 2016 database and the experimental setup. Section 4.5 reports the results obtained. A summary of the work and findings is presented in Section 4.6.

4.1 Introduction

Feature extraction (FE) is a step common to most machine learning applications. In voice biometrics, audio signals are treated to generate hand-crafted acoustic feature vectors that highlight information relevant to the final application, e.g. characteristics

of the vocal tract. Despite recent attempts reported in the literature to design features automatically, i.e. without exploiting human knowledge, using so-called end-to-end systems [156], results show that these techniques are still not mature. At least until they are, hand-crafted FE continues to be a fundamental stage.

Acoustic features are usually derived from spectral analysis and short-time Fourier Transform (STFT). Nonetheless, alternatives motivated by multi-resolution time-frequency analysis employed in music processing, and particularly the constant Q transform (CQT) [157, 158, 159], have provided scope for research in numerous speaker-identity related tasks. The CQT provides a higher frequency resolution at lower frequencies. These frequencies normally portray higher harmonic densities, and thus a higher resolution around them may benefit the analysis of voiced speech content [160]. At the same time, the CQT provides high time resolution at higher frequencies, allowing for rapid change detection in such ranges. Results in the literature have reported the successful application of CQT-derived features to speaker anti-spoofing [161, 162] and utterance verification [160].

On a different line, the use of BKs for SD has been subject to a number of improvements in recent years (see Section 3.6.2). These explored enhancements to the different blocks that compose the traditional pipeline (Section 3.7), such as alternatives to KBM composition or clustering selection alternatives, all seeking to improve upon the original implementation proposed in [10]. In doing so, all of these works relied on traditional acoustic feature extraction (MFCCs), without analysing the relationship between acoustic features and the BK speaker modelling technique.

Traditionally, SD algorithms use models trained specifically for speaker recognition with large amounts of background data. The leveraging of this data can help to mitigate the effects of within-speaker variability. In contrast, BK speaker modelling for SD (see Section 3.2.2), does not rely on any external training data. It uses exclusively the speech content available in the test session, also in the form of acoustic features, to generate speaker-discriminative representations. The influence of the acoustic features is particularly important in especially in the case of two fundamental processing stages. First, to compose the KBM, a Gaussian pool is filled with Gaussian distributions which are fitted directly to the in-session acoustic features using a sliding window (approach described in Section 3.2.2). Components in this Gaussian pool are iteratively chosen to compose a KBM that minimizes redundancy while maximizing speaker discriminability. The success of this process is strictly related to the acoustic features and their discriminative capacity. Second, the acoustic features are compared to the components in the KBM by calculating their individual likelihoods, in order to project them into the binary domain via a top-M selection. This straight-forward operation upon only in-session acoustic features ensures

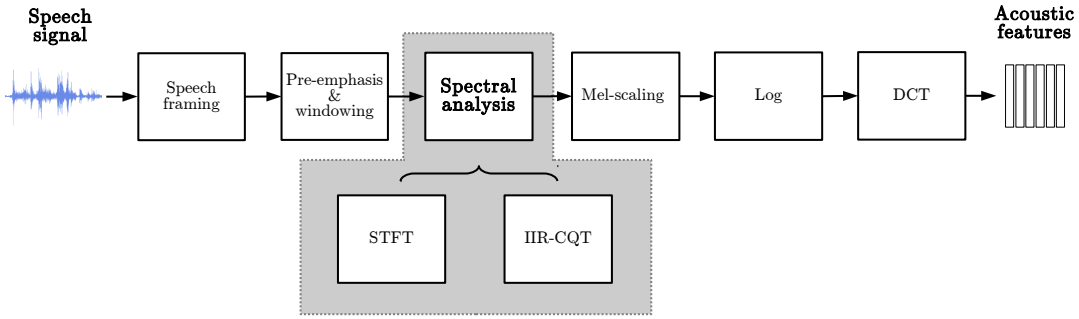


Figure 4.1: Traditional feature extraction pipeline for speech processing. The spectral analysis block is the subject to study in this chapter. Short-Time Fourier Transform (STFT) is usually employed and is characterised by representing the spectrum at a constant resolution. By contrast, Infinite Impulse Response - Constant Q (IIR-CQT) spectral analysis provides with a multi-resolution time-frequency alternative.

that BK speaker modelling is robust to domain variation, simply because it makes no use of out-of-session, external data. However at the same time, it makes of within-session acoustic features variability a major problem for BK-based SD, which could arguably be mitigated via enhanced spectral analysis.

The promising results in other related applications, and the strong relationship between BK speaker modelling and acoustic features motivate the main contributions of the work reported in this chapter. An analysis of the impact of the time-frequency resolution research used in FE upon the BK speaker modelling technique is reported. The work considered the very first assessment of CQT-inspired features paired with traditional Mel-cepstral analysis, introduced in [160], for SD and BK speaker modelling.

The work reported in this chapter relates to EURECOM's participation in the Albayzin 2016 Speaker Diarization Evaluation [12]. The Albayzin evaluation series promotes research on a number of speech processing tasks, such as audio segmentation, SD, text-to-speech, language recognition and spoken term detection. It is organised by the Spanish Thematic Network on Speech Technologies. The evaluation provided an ideal opportunity to assess and compare the benefit of the BK approach, and the multi-resolution spectral analysis when pitched against competing systems proposed by other leading research laboratories worldwide.

4.2 Spectral analysis

A typical pipeline for FE in speech processing tasks is presented in Figure 4.1. This chapter explores the influence of spectral analysis (dark grey box in Fig. 4.1) and, more specifically, the dependence upon spectral resolution, upon BK speaker modelling while

the remainder of the pipeline remains unchanged. This section, describes the differences between the two spectral analysis methods employed in this work, namely the Short Time Fourier Transform (STFT) and the Infinite Impulse Response-Constant Q Transform (IIR-CQT).

4.2.1 Short-time Fourier transform

The STFT is a basic spectral analysis tool widely applied in audio processing for a number of applications related to speech and music. Given pseudo-stationary segments of an audio signal to which the Fourier transform can be applied, STFT analysis can be used to derive an estimation of its spectrum. The time-frequency resolution with which the STFT represents the spectrum is determined by the length in samples of the employed window N_s. This length establishes the number of bins $N_k \approx N_s$ in which the spectrum is represented (where the number of bins is usually increased to the next power of 2 which is greater than the number of samples N_s in order that the STFT can be estimated using the Fast Fourier Transform (FFT)). In general, a bin $k \in \{1, \ldots, N_k\}$, has center frequency $f_k = k \cdot \frac{f_s}{N_k}$, where f_s is the sampling frequency. Bins are positioned linearly in frequency separated by $\Delta_f = \frac{f_s}{N_k}$ Hz. STFT-derived spectra thus have fixed frequency resolution. Conceptually, this is also equivalent to a filter bank with a varying Q factor. The Q factor can be defined as a measure of filter selectivity or quality factor for a certain frequency f_k. In the case of the STFT, $Q_k = f_k/\Delta f$. This fixed frequency resolution assumes the equal importance of the different frequencies of the spectrum in the resulting spectral analysis, i.e. they are analysed with equivalent precision. For voiced speech signals, relevant information located within the lower range of frequencies is emphasized by means of the application of Mel-scaling (following spectral analysis as visible in Fig. 4.1). However, these differences are not reflected in the frequency resolution of the STFT itself, thus possibly biasing the result of Mel-scaling. An alternative *default* spectra representation that captures changes in low frequencies without sacrificing time resolution might thus be desirable. These limitations in spectra representation consequently motivate the exploration of spectral analysis beyond its fixed resolution as introduced in the following sections.

4.2.2 Multi-resolution time-frequency spectral analysis

Alternatives to traditional spectral analysis have been widely explored in the literature in the context of music processing [163, 164], where the density of the harmonics is not linearly distributed along the spectrum. Multi-resolution spectral analysis has not attracted research interest in voice biometrics until recently. The constant Q transform [157] was recently explored in the context of speaker anti-spoofing [161, 162].

Meaningful contributions were also reported in the literature in other related tasks such as text-dependent speaker recognition [160].

Constant Q transform

Originally introduced by Youngberg and Boll [157] followed by the work of Brown [158], the constant Q transform (CQT) offers a multi-resolution time-frequency alternative to more traditional spectral analysis. The CQT has non-linear spectral resolution where the Q factor, introduced in Section 4.2.1, is set to:

$$Q = \frac{f_k}{\Delta f_k}. \tag{4.1}$$

In contrast to the linear scale of the STFT, where $Q \propto f_k$, here the Q factor is fixed to a constant value. With constant Q, the spectral representation mimics that of the Q factor of the human hearing system, which is known to be pseudo-constant for frequencies between 500Hz and 20kHz [165]. In the CQT, this is possible thanks to the variable bandwidth of the different bins Δf_k, which are now also dependent on f_k. The central frequencies f_k are thus distributed geometrically instead of linearly as in the STFT. In consequence, consecutive bins at lower frequencies are closer together than bins at higher frequencies. The result is a spectro-temporal decomposition with a higher spectral resolution at lower frequencies and a higher temporal resolution at higher frequencies. In this result, however, the geometric spacing of the central frequencies employed to model the spectrum in the CQT means that further undesired processing is needed to decorrelate the output before the application of traditional Cepstral analysis [161, 166], is common in voice biometrics.

Infinite impulse response - constant Q transform

Direct evaluation of the CQT implementation as in [158] is very time consuming. In consequence, developments in CQT analysis have resulted in implementations that attempt to minimize these computational needs, e.g. the authors in [167] proposed an efficient implementation that leverages the Fast Fourier Transform (FFT), as per implementation used in [158]. In another work [168], different authors proposed a bounded-Q transform (BQT) that combines the FFT with a multirate filterbank. The work explored in this chapter follows another FFT-based approach to CQT modelling reported in [159], that formulates the computation of the CQT as the task of designing an Infinite Impulse Response (IIR) filterbank.

The work in [159] proposes an alternative method to calculate the CQT that is

computationally simpler than that in [167], while being flexible regarding constant Q design criteria. To achieve the time-frequency multi-resolution that characterises the CQT, the filters in the filterbank have different impulse responses for different frequencies. These filters are applied directly to the FFT and, consequently, represent the spectrum on a linear scale, as per the geometrical scale obtained by the conventional CQT [158]. Following the notation in [159], the k^{th} filter in the filterbank is defined as a first order IIR filter:

$$Y_k[n] = X[n] - z_k X[n-1] + p_k Y_k[n-1] \tag{4.2}$$

where $X[n]$ is the DFT of the signal, with pole p_k and zero z_k. Such filters must be calculated for every $k = \{1, \ldots, N_k\}$ where N_k is the number of bins in the DFT. In this sense the computational cost of the method proposed in [159] would still not deliver any improvement in terms of computational cost over that in [167]. Some assumptions are proposed in [159] to improve efficiency. Choosing a different filter response of the IIR filterbank for every frequency bin can be considered as applying an Linear Time Variant (LTV) system to the DFT of a frame. The desired response of the LTV for a given frequency bin is the impulse response of the correspondent filter. If the LTV system changes slowly over time, its Steady State Response (SSR) can be implemented by means of a single Time Varying (TV) IIR filter. This simplification, acceptable in the case of constant Q-motivated spectral analysis (where, by using varying windowing to obtain a constant Q, time windows for two consecutive frequency bins are expected to be highly related), considerably reduces the number of computations. In consequence, the CQT-derived spectrum of the DFT of a frame $X[n]$ can be formulated similarly to that defined in Equation 4.2. It differs in that the poles are now defined as a variable dependent on the frequency $p = p[n]$ and can be defined as:

$$Y[n] = X[n] - X[n-1] + p[n]Y[n-1]. \tag{4.3}$$

The pole variable $p[n]$ must therefore be computed only once, thereby simplifying the process. But this simplification does not come free of cost, as it implies a decreased precision in maintaining a constant value of Q along the spectrum, which is the ultimate goal of CQT spectral analysis. To counter this discrepancy the authors in [159] introduced a compensation technique that allows for the Q factor to remain almost perfectly constant. The design of these compensated IIR filters is not a subject of study in this thesis and is therefore left as further reading. Full details are presented in the original work [159].

In conclusion, the IIR-CQT provides an alternative to CQT computation [167] that is computationally efficient and operates using a linear scale by leveraging the FFT computation, while maintaining a constant Q factor. The use of a linear scale improves on processing efficiency by avoiding to resample the spectrum from geometric to

linear [161, 166]. This allows for further ceptral analysis to be applied in straightforward manner.

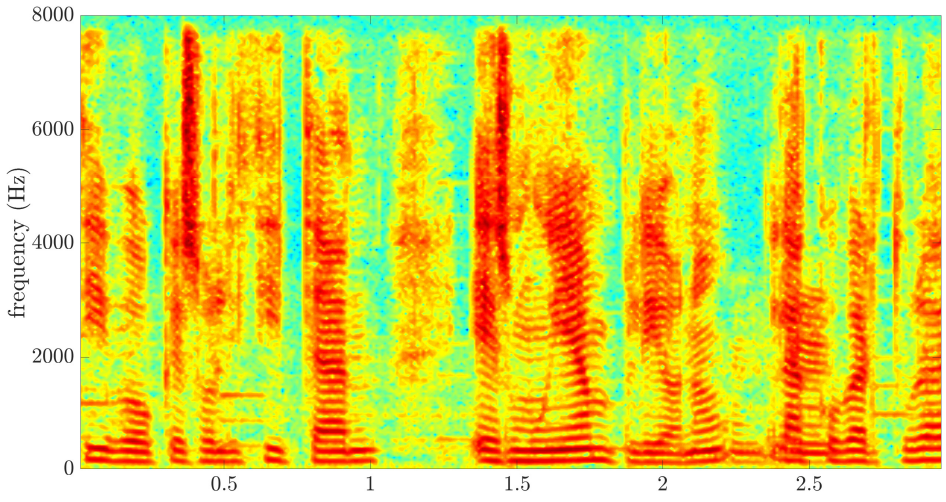
From spectral to cepstral analysis

As described in Section 4.1, the analysis proposed in this chapter is motivated by the work in [160] in which the use of the IIR-CQT was proposed for utterance and text-dependent speaker verification. Thanks to the linear scale produced by the IIR-CQT the usual steps that follow spectral analysis can then be applied as before. These are illustrated in Figure 4.1, i.e. Mel-scaling, log computation, and decorrelation by means of the DCT for cepstral analysis (see Chapter 2). The features that result from this processing were named **I**nfinite impulse response **C**onstant Q **M**el-frequency **C**epstral coefficients (ICMC) in [160] and are tested in the following sections.

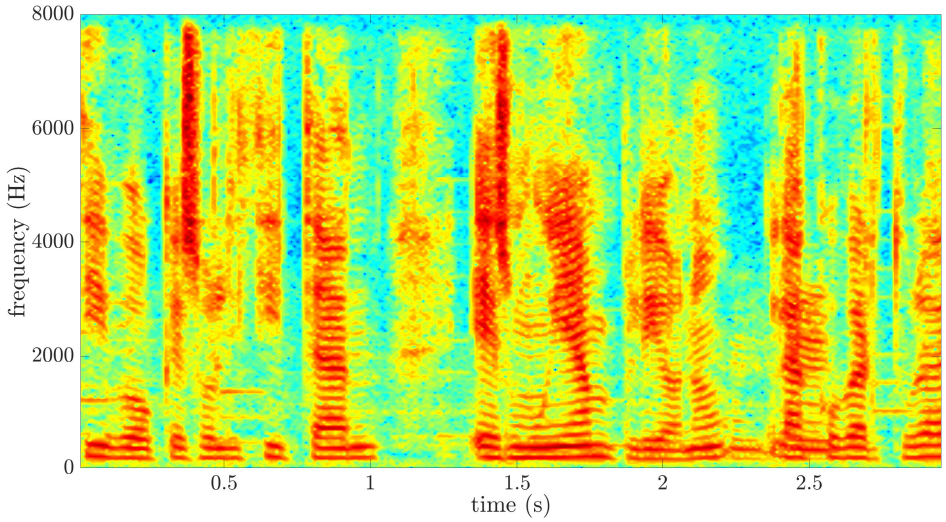
4.3 Proposed analysis

To assess the influence of alternatives to the STFT to BK speaker modelling, two spectral analysis techniques alternatives are considered. The baseline system uses traditional MFCC features derived from conventional STFT spectral analysis. Results for this system are compared to those for an identical system that uses CQT spectral analysis. and constitute our baseline. The research hypothesis is that the higher resolution at lower frequencies could give ICMCs an advantage over MFCCs upon their application to BK speaker modelling. On the other hand, it could be argued that the resolution achieved by ICMCs in lower frequencies could be similarly replicated by means of MFCCs extracted using a very high but constant resolution. Only then could it be affirmed that a constant-Q-derived mapping of the spectrum is the cause to the increase in performance reported in literature [160, 161, 166].

This analysis can be achieved in a rather straight-forward manner by generating spectra representations using time frames of different lengths. As described above, the length of the time window N_s used to frame the speech signal usually defines the number of bins into which the spectrum is divided. We consider $N_k = N_s$ for the sake of direct comparison between the proposed methods, and do not perform zero-padding. Normally, MFCCs are extracted from pseudo-stationary frames of speech whose length is in the order of 25ms. In an audio file with a sample frequency of 16kHz, like that used in [160], the resulting number of bins is $N_k = 25\text{ms} \times 16\text{kHz} = 400$ (normally rounded up to 512 bins). ICMCs, as in [160], are extracted from a speech window that generates a number of bins $N_k = 2048$. In time, that is equivalent to an unusually large window of 128ms, in which the pseudo-stationary quality of the speech is questionable. However,



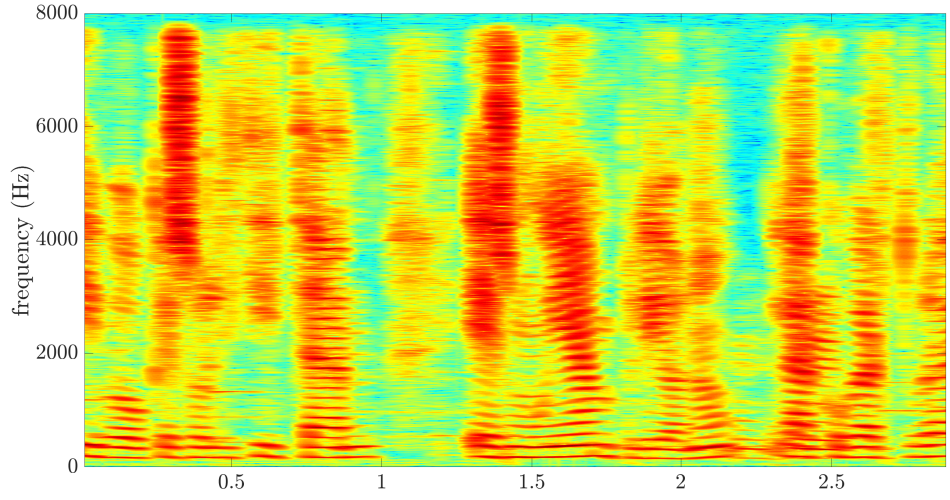
(a)



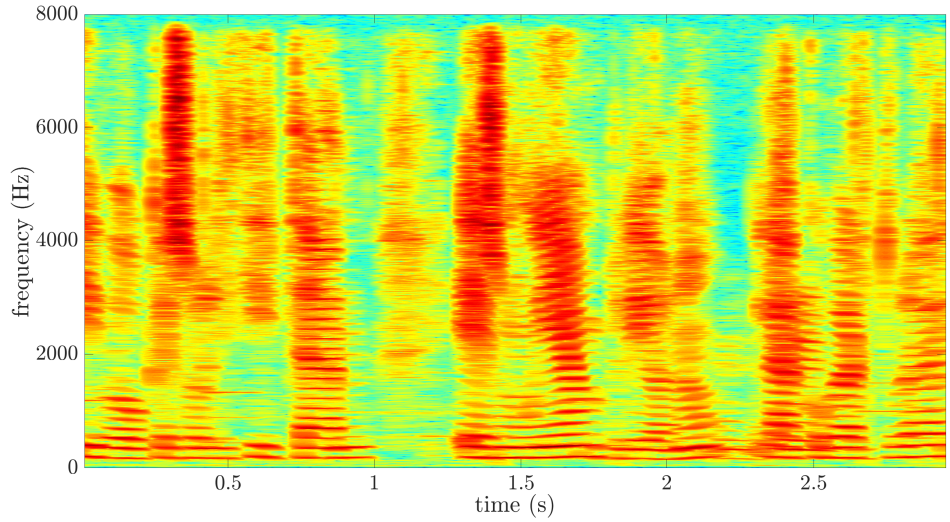
(b)

Figure 4.2: Spectrograms of a 3s speech segment extracted from an audio file in the Albayzin 2016 database [12]. Spectrograms computed using a window length of 25ms and $N_k = 400$ bins for the (a) STFT and (b) IIR-CQT.

such a window length allows for frequency resolution to be very high. The extent to which the constant-Q factor influences performance, is unknown. MFCCs extracted with windows of $N_k = 2048$ bins could provide with a similarly high resolution at low frequencies without the need for the further processing necessary to obtain the IIR-CQT. We are interested in analysing the effectiveness of the constant-Q spectral analysis of ICMCs against the STFT of traditional MFCCs in an unbiased comparison that is not



(a)



(b)

Figure 4.3: Spectrograms of a 3s speech segment extracted from an audio file in the Albayzin 2016 database [12]. Spectrograms computed using a window length of 128ms and $N_k = 2048$ bins for the (a) STFT and (b) IIR-CQT.

conditioned by resolution.

The comparison is performed with MFCC and ICMC features extracted from time windows of both 25ms and 128ms with, respectively, low and high frequency resolutions. Spectrograms derived using 25ms windows and both STFT and IIR-CQT spectral analysis are illustrated in Figure 4.2. Figure 4.3 illustrates the same, except for 128ms windows. The baseline system uses MFCC features extracted from a spectral analysis based on

$N_k = 400$ bins and a STFT decomposition (Fig. 4.2a). Similarly for the ICMCs, 400 bins are derived using the IIR-CQT (Fig. 4.2b). For a 128ms window length, MFCCs are extracted with $N_k = 2048$ linearly spaced bins, providing a much higher resolution, that is fixed along the spectrum (Fig. 4.3a). For ICMCs, IIR-CQT generates a similarly high resolution at low frequencies, that shifts towards a high time resolution at higher frequencies (Fig. 4.3b). Upon first sight, the expected differences in terms of time-frequency resolutions are only evident between the standard implementations of STFT (25ms window length in Fig. 4.2a) and IIR-CQT (128ms window length in Fig. 4.3b [160]). When extracted using equivalent configurations, both STFT and IIR-CQT spectrograms are clearly similar to each other, to the point where it is hard to tell if there is any difference at all, despite the possible effect of the IIR-CQT. The fact that high resolutive spectrograms derived from both STFT (Fig. 4.3a) and IIR-CQT 4.3b) are so similar arises the question as to the contribution of the ICMCs with regard to MFCC features as reported in [160]. The relevance of these visually imperceptible differences is put to test in experiments that explore their impact in terms of BK speaker modelling.

4.4 Experimental setup

In this section, the approach to compare the relative benefit of STFT and IIR-CQT spectral analysis for BK speaker modelling is described. Section 4.4.1 describes the database provided for the Albayzin 2016 Speaker Diarization Evaluation that was used in the experiments reported here. In Section 4.4.2 further details are given for feature extraction. Section 4.4.3 describes the configuration used for the BK speaker modelling module. The relative speaker discriminability of STFT- and IIR-CQT-derived CVs using BK speaker modelling is assessed in two experiments: Section 4.4.4 describes the configuration of a controlled speaker recognition experiment, whereas Section 4.4.5 explains that of a SD experiment. Note that, for all experiments in which a VAD system is required, oracle annotations are used.

4.4.1 Database

The dataset employed was provided in the context of the Albayzin 2016 Speaker Diarization Evaluation. Audio files from various origins constitute the different data subsets for training, development, and testing. The training set, obtained from the Catalan broadcast news database from the 3/24 TV channel, was already used for the 2014 Albayzin Audio Segmentation Evaluation. It was recorded by the TALP Research Centre from the UPC in 2009 under the Tecnoparla project [169]. The database contains approximately 87 hours of recordings of which speech in Catalan language constitutes approximately a 92%. They

are labelled at the music and noise segmental level annotation which imply, respectively, a 20% and a 40% of the time. Finally, overlap is present in two different ways. 40% of speech time is overlapped with noise meanwhile 15% is overlapped with music. The development and test sets are composed of files donated by the Corporacion Aragonesa de Radio y Television (CARTV). A total of approximately twenty hours selected from the Aragon Radio database are split into two groups. One contains four hours of data and comprises the development set, whereas the test set is composed of the remaining sixteen hours of data. Regarding its content, this second dataset is composed of approximately 85% speech in Spanish language, 62% music and 30% noise, where overlap is distributed as follows: 35% of the audio contains music along with speech; 13% overlaps speech with noise; a 22% contains speech alone. All data is supplied in PCM format, mono-channel, little endian encoding 16 bit-per-sample and with a 16 kHz sampling rate. It is interesting to note that training set is provided in a different language than development and test sets. As BK speaker modelling allows for an in-session modelling of the acoustic space, the technique is able to overcome this limitation for data-dependent speaker modelling algorithms. In consequence, results are reported on the development and test set without any use of the training set.

4.4.2 Feature extraction

Features employed in the experiments reported below were extracted using frame lengths of 25ms ($N_k = 400$ bins for both STFT and IIR-CQT) and 128ms ($N_k = 2048$ bins). MFCCs use a standard STFT implementation whereas ICMCs use the CQT implementation reported in [160][1] and an empirically optimised Q-factor of $Q = 96$ [160, 161]. Besides those differences, they share the following common configuration parameters: 19 static cepstral coefficients are extracted with a pre-emphasis factor of 0.97, using a frame shift of 10ms, and a 20-channel Mel-scaled filterbank.

4.4.3 BK speaker modelling configuration

The details of the BK speaker modelling configuration follow the notation used in Section 3.7. For the KBM training a 2s window with a shift of 0.5s is used to train the initial Gaussian pool with a minimum number of Gaussians $G = 1792$. As regards to the final KBM size, it is selected as a percentage of the initial pool size. In this way, the model size is chosen adaptively with regard to the audio file duration. The relative KBM size N is swept across different percentages that go from $\alpha = 5\%$ to $\alpha = 100\%$ of the initial Gaussians sampled from the audio, in order to find the best configurations. In this work, results are reported in terms of cumulative vectors (CVs). The computation

[1] Code to replicate these features is available for download at: http://audio.eurecom.fr/content/software

of CVs from input data is performed using segments of 3s. The number of top Gaussians per frame $M = 5$ is set to 1 for feature binarization.

4.4.4 In-session speaker recognition

A first experiment in the form of a controlled speaker recognition scenario is proposed, in order to measure the impact of the different spectral analysis variations in the final speaker discriminative capacity of the CVs. The similarity matrix of the CVs extracted for each session in the development dataset is calculated, and the resulting scores are pooled. Given a cosine distance metric the scores range between 0 and 1 so the scores are already normalized and comparable across sessions. It is important to note though, that cross-session comparisons between CVs are not possible in this experiment due to the use of session-dependent KBMs. By using the oracle session annotations, trials are labelled as target (CVs belonging to the same speaker) or non-target (CVs belonging to different speakers). Speaker recognition performance is reported in terms of the equal error rate (EER, see Chapter 2) expressed as a function of the KBM size N, which depends on the percentage α of the number of components G in the complete Gaussian pool.

Two scenarios are analysed. First, segment CVs are compared against each other in a short-utterance versus short-utterance speaker recognition experiment (CVs are extracted from 3s speech segment). Second, segment CVs are compared to speaker cluster CVs. Cluster CVs are extracted in an oracle manner from the full pool of speech data available for each given speaker in a session. This second scenario intends to simulate the behavior of the CVs in a long-utterance (cluster speaker CVs) vs short-utterance (segment CVs) speaker recognition scenario. The results from this experiment should allow to visualize the influence of the different spectral analysis alternatives and configurations without the influence of the complete pipeline of a SD system.

4.4.5 Speaker diarization experiment

Performance is also measured by means of a fully fledged SD pipeline similar to that established as the baseline as described in Section 3.7. Segment level CVs are clustered through a bottom-up AHC algorithm with segment-to-cluster reassignment allowed at each iteration. As for clustering initialisation, $N_{init} = 25$ initial clusters are derived from speech chunks of equal size. This number is related to the maximum number of speakers found in the audio files of the development set (16 speakers at most, hence setting $N_init = 25$ to allow for the AHC algorithm to converge). Performance is reported in terms of the diarization error rate (DER, see Chapter 2), evaluated with a standard forgiveness collar of 0.25s. The DER is first calculated for solutions in which the number

of speakers is obtained automatically by means of an elbow criterion [149]. Results for the solutions with the number of speakers that generate the lowest DER chosen per session in an oracle manner are also reported.

4.5 Results

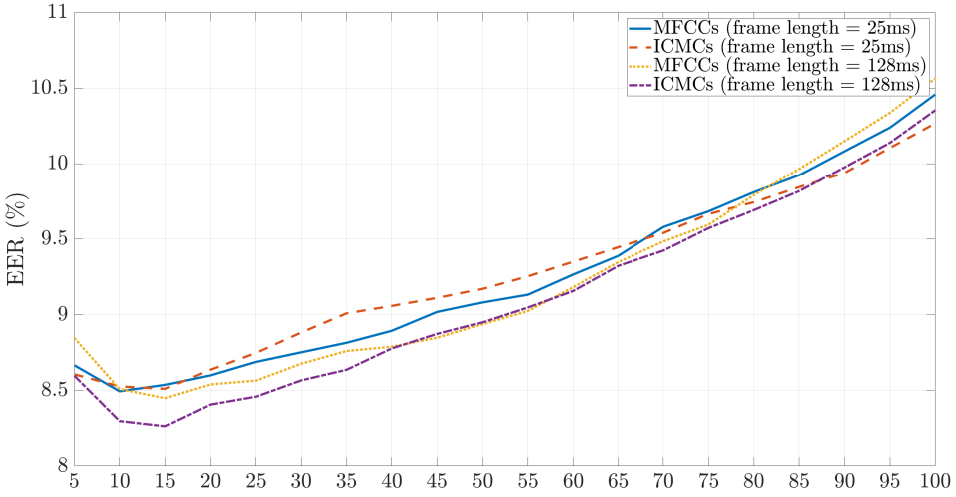
This section reports the results for the two different experiments described above. They serve to compare the relative benefit of features derived from both STFT (MFCCs) and IIR-CQT (ICMCs). Section 4.5.1 discusses the results in terms of speaker recognition whereas Section 4.5.2 describes results for SD. All results reported correspond to experiments performed on the development set of the Albayzin 2016 Speaker Diarization Evaluation dataset. Section 4.5.2 also includes results on the test set for our final submission to the challenge.

4.5.1 Speaker recognition

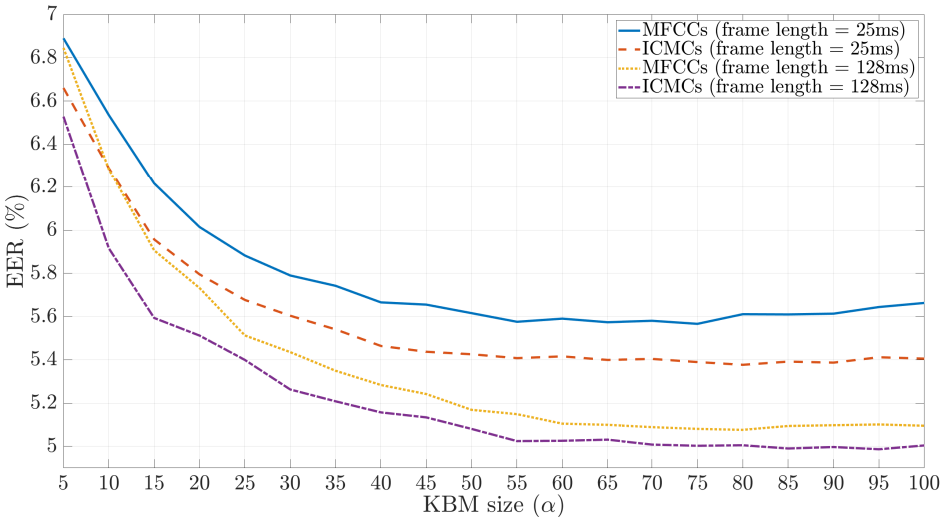
Results obtained for the speaker recognition experiment are illustrated in Figure 4.4. The EER is plotted as a function of the varying size of the KBM α, defined as a percentage of the size G of the Gaussian pool.

Figure 4.4a shows the EER obtained for segment CVs extracted from segments of 3s of speech (short-short condition). Use of MFCCs (solid blue line) extracted over 25ms constitute the baseline which delivers an EER in the order of 8.5% for a KBM size of $\alpha = 10\%$. When ICMCs (dashed orange line) are extracted over 25ms, the performance achieved is not better than that of MFCCs for most of the KBM sizes. For a KBM size of $\alpha = 15\%$ performance is similar to that of the MFCCs is reached.

Use of MFCCs extracted over 128ms (dotted yellow line) deliver a slight improvement over their 25ms counterpart. On the other hand, ICMCs extracted using 128ms-long frames (dash-dot purple line) improve upon both the performance of the baseline and MFCCs extracted over the same 128-ms-long frames. For a KBM size of $\alpha = 15\%$ the EER is 8.3%. It is important to note that while the use of ICMCs extracted over 128ms frames reportedly delivers the best performance out of the evaluated front-ends when comparing pairs of CVs extracted over 3s of speech data. However, this comparison, i.e. short CV vs. short CV, is never performed within our SD pipeline, which operates on a segment-to-cluster approach to AHC in which clusters are initialised using larger amounts of speech. We are consequently more interested in such a scenario and results for it are presented in the next paragraph.



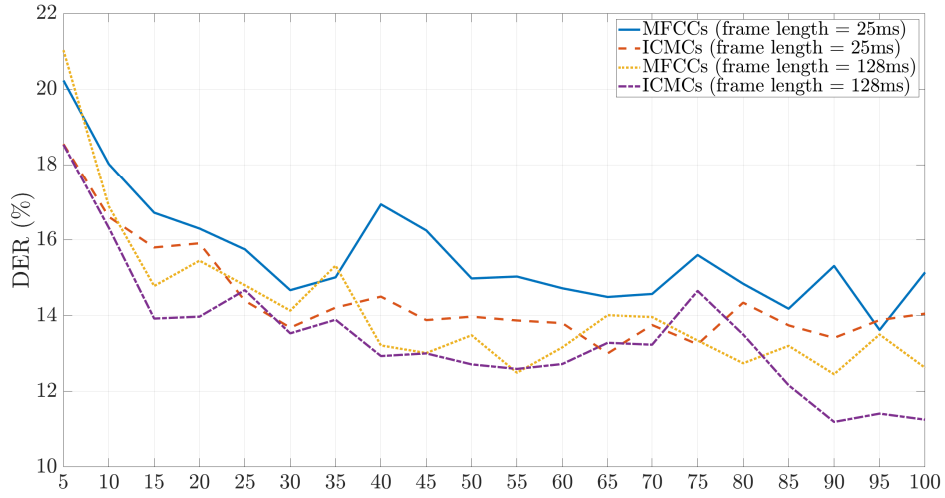
(a)



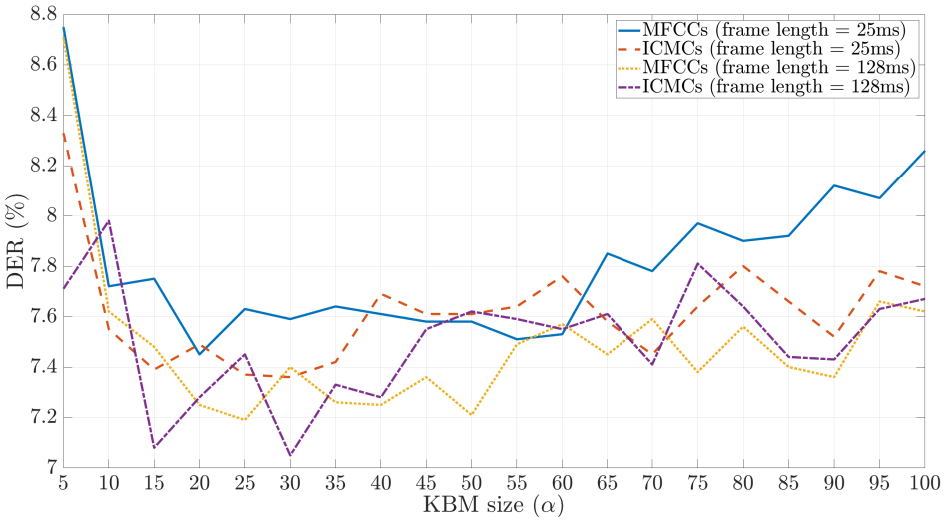
(b)

Figure 4.4: Performance is measured in terms of equal error rate (EER,%) for different KBM sizes (α). Results in (a) are for comparisons between CVs extracted from 3s speech excerpts, while in (b) CVs extracted in 3s are compared against CVs modelled on all the available speech in the oracle speaker clusters.

Figure 4.4b portrays a comparison between CVs, similar to the way they are processed in our SD pipeline. CVs modelled on 3s speech segments are compared against the speaker-wise CVs contained within a session (short-long condition), which are extracted using the oracle annotations. The use of oracle annotations allow us to simulate a perfect initialisation of the clusters that are available when the AHC algorithm starts operating. MFCCs extracted over 25ms (solid blue line) reach their optimal performance for KBM



(a)



(b)

Figure 4.5: Diarization performance in terms of diarization error rate (DER) as a function of the KBM size (α). Systems used in (a) and (b) differ in the method employed to determine the number of speakers per session. In (a) they are determined by means of the baseline method based on an the elbow criterion, while in (b) the number of speakers is determined in an oracle manner.

sizes between $\alpha = 55\%$ and $\alpha = 75\%$, with an EER = 5.6%. Here, ICMCs extracted over 25ms of speech (dashed orange line), offer a consistent improvement of roughly 0.2% EER over that of MFCCs. When features are extracted over 128ms of speech, both MFCCs (yellow dotted line) and ICMCs (dash-dot purple line) deliver results in which the differences are even more evident as with regard to using features extracted

over 25ms frames. MFCCs bring the EER to around 5.1% for KBM sizes over $\alpha = 90\%$. ICMCs give an slightly lower EERs of below 5% for similarly sized KBMs. The difference between MFCCs and ICMCs is small but consistent when CVs are modelled over amounts of speech of sufficient length. Segment-level CVs are a challenging representation as they are extracted over small amounts of speech data, however, the robustness of the cluster-level CVs allows for clearer, more consistent improvements.

The results in these two experiments allow to draw some interesting conclusions related to questions posed in Section 4.3, as to assessing the effect of the resolution separately from that of the constant or variable character of the Q factor. First, BK speaker modelling benefits from features extracted using frame lengths that are longer than those typically used in speech processing, e.g. features extracted from 128ms frame lengths perform better than those extracted from 25ms-long frames. In short-short (Fig. 4.4a) and short-long conditions (Fig. 4.4b), MFCCs and ICMCs extracted from 128ms of audio outperform features extracted from 25ms. This allows us to affirm that the higher frequency resolution is of benefit to BK speaker modelling for both variable Q (that of STFT) and constant Q transforms (that of IIR-CQT). At the same time, it is also shown that frequency resolution is not the only factor responsible for this improvement. The constant Q mapping of the spectrum of the ICMCs, characterised by a higher time resolution for higher frequencies w.r.t MFCCs extracted by equivalent frame lengths, provides with a 12% relative increase in performance over the 25ms MFCC baseline as w.r.t the 9% achieved by 128ms MFCCs. This justifies the usage of ICMCs for BK speaker modelling over that of MFCCs, particularly in the case of frame lengths that allow for higher resolutions.

4.5.2 Speaker diarization

In this section, two different SD experiments are reported. These vary in terms of the method which was employed to determine the number of speakers per session over the development set. The objective of these experiments is to assess the effect of the different alternatives to spectral analysis in the context of a complete SD pipeline. Some preliminary conclusions can be drawn from the results presented in Section 4.5.2 as to the effect of the different spectral analysis techniques to BK speaker modelling. However, it is of our interest in this thesis to evaluate the impact of such front-ends within a SD system.

Results in SD are reported in terms of DER in Figure 4.5. Following the presentation format of the previous results in Section 4.5.1, the DER is plotted as a function of the KBM size α. The DER is calculated with a standard 0.25s forgiveness collar.

Automatic speaker number estimation

Results presented in Figure 4.5a are obtained by estimating the number of speakers automatically. This is done by means of a WCSS-dependent elbow criterion [149], a component of the baseline pipeline defined in Section 3.7. When extracted from frames of 25ms, ICMCs (dashed orange line) reduce the DER by a relative average of 7% compared to MFCCs (solid blue line) for almost all KBM sizes. On the other hand, using 128ms frame lengths, performance is increased for both approaches, with ICMCs offering a consistently better result, with a bigger difference between ICMCs (dash-dot purple line) and MFCCs (dotted yellow line) for larger KBM sizes. These results confirm the findings observed in Section 4.5.1, where the extraction of features from frames of length 128ms deliver consistent performance benefits to in BK speaker modelling. In this case, SD performance, despite its more complex system pipeline, is improved by a consistent 2% DER for most KBM sizes. In particular, for larger KBM sizes for $\alpha > 90\%$, the DER decreases by over 5% DER corresponding to a relative improvement of nearly 30%.

Oracle speaker number estimation

In order to evaluate the AHC algorithm without the influence of the method used to estimate the number of speakers, results for a second set of experiments in SD are reported in Figure 4.5b. Here, the number of speakers per session is estimated in an oracle manner, by selecting the number of clusters that yield the minimum DER per session. This is, of course, unfeasible in a real scenario, but facilitates the assessment of the proposed spectral analysis solutions in the context of a controlled AHC process. Results for 25ms frame lengths are reported also here for MFCCs (solid blue line) and ICMCs (dashed orange line). Despite the inconsistent tendency generated by the varying size of the KBM, ICMC features lead to a lower DER than that obtained using MFCCs for most KBM sizes. For results obtained using frame lengths of 128ms, ICMCs (dash-dot purple line) and MFCCs (dotted yellow line) experiments report similar performances that outperform that of the 25ms frame lengths. Nonetheless, ICMCs produce a minimum overall DER of under 7% for KBM sizes of $\alpha = 30$.

These two experiments confirm the tendencies observed in Section 4.5.1 despite the inclusion of the AHC processs. On the other hand, it is interesting to observe that the optimum KBM sizes do not align for automatically estimated (Fig. 4.5a) and oracle (Fig. 4.5b) speaker numbers. While the reason behind this difference in behavior is not explored in this thesis, it would be of interest to explore in future research. Figure 4.5b shows that the best achievable performance is reached using small KBM sizes with $10 < \alpha < 30$. Comparatively smaller KBMs allow to derive segmental representations that are also of smaller dimension, allowing for lighter computations of datasets. This

		EURECOM [155]	Team 2 [170]	Team 3 [171]	Team 4 [172]
DER (SER) (%)	Development	**11.9 (9.4)**	-	16.2 (11.7)	27.0 (18.4)
	Test	**18.2 (13.9)**	18.3 (14.0)	25.7 (17.0)	32.6 (20.0)

Table 4.1: Results of the Albayzin 2016 Speaker Diarization Evaluation for all 4 participants, in both development and test sets. Please note that Team 2 and our submission used oracle annotations for VAD, while Team 3 and Team 4 employed an automatic approach. An exact comparison between both approaches is not possible, but to make for a more fair comparison, speaker error rate (SER)(DER excluding VAD-derived errors) is also reported.

motivates the search for better stopping criterion mechanisms for AHC that allow to correctly estimate the number of speakers for such KBM sizes.

Results in the Albayzin 2016 Speaker Diarization Challenge

The work and enhancements explored in this chapter were tested against other SD systems in the unseen test set (belonging to the same domain as the development set) of the Albayzin 2016 Speaker Diarization Evaluation dataset. This context allowed to evaluate the performance of a BK SD system using ICMC features against other SD pipelines, with all details of the EURECOM submission available in [155]. The results are presented in Table 4.1, for both development and test sets for our submission (EURECOM[2]) and the 3 other participants. It is important to note that while our submission and that of Team 2 [170] employed oracle VAD annotations, Team 3 [171] and Team 4 [172] used an automatic VAD system. To be able to compare for each system in a consistent manner while excluding errors derived from VAD systems, speaker error rate (SER) is reported together with the DER (which still does not make for a direct comparison, as automatic VAD may also increase SER). Details of the other competing systems are as follows:

- **Team 2 [170]**: A Bayesian information criterion (BIC) blind speaker segmentation followed by an AHC approach based on i-vectors.
- **Team 3 [171]**: BIC segmentation followed by an online clustering of i-vectors using PLDA scoring.
- **Team 4 [172]**: BIC segmentation followed by clustering based on GMMs and Viterbi alignment.

[2]Please note that the performance reported here differs slightly from that achievable in the development set as reported in this chapter. This is due to small improvements having been applied in the overall structure of the system, which nonetheless respect the trends reported in [155] and here. Results are reported as in [155] for comparison with results for other participants.

It is interesting to observe that Team 2 and Team 3 systems are based on i-vectors, a technique heavily dependent on external training data. Team 4 follows an approach similar to that reported in this thesis in that no external training data is needed. The variation in approaches allows us to make some observations. First, that the use of external training data is not always beneficial. When confronting a closed-set training condition (like that for which results are reported), the lack of extensive in-domain training data can be a burden when suitable domain adaptation techniques are not applied or are simply not feasible. Unless the data requirements to make those data-hungry techniques perform in a reasonable manner are used, BK speaker modelling offers a robust and computationally light approach. Second, the superior performance of our BK speaker modelling and ICMC-based approach to that of Team 4 highlight the benefit of employing a KBM to model the local acoustic content over that of traditional GMM-HMM and Viterbi based methods. The results in Table 4.1 show that our system achieved the best performance on test set of this evaluation, justifying the further development of techniques and improvements applicable within its training-data independent nature.

4.6 Summary

The work presented in this chapter proposes an alternative to traditional spectral analysis as a method to improve the discriminability of BK-based speaker representations. Acoustic features are traditionally extracted in the form of Mel-frequency Cepstral Coefficients (MFCCs), which rely on spectral analysis based on the short-time Fourier transform (SFTF). In this chapter, an alternative multi-resolution spectral analysis tool is explored based on the constant Q transform (CQT), which has shown benefits in performance in other speech processing related tasks. In particular, the infinite-impulse response, constant Q transform (IIR-CQT) Mel-frequency cepstral (ICMC) coefficients, recently developed by other authors [160], are assessed in their first time application to BK-based speaker modelling. Results are reported in terms of speaker recognition and SD for various front-end optimisation experiments. These results highlight the positive impact of multi-resolution spectral analysis on the discriminability of BK speaker modelling: an ICMC-based front-end led to a relative improvement in DER of 14% over the MFCC-based baseline. These substantial improvements led to a competitive performance of the resulting, enhanced BK-based SD system in the submission made to the Albayzin 2016 Speaker Diarization Evaluation, in which it obtained 1st place.

Chapter 5

Speaker change detection with contextual information

Speaker change detection (SCD) can be of benefit to a number of different speech processing tasks such as speaker diarization (SD), recognition and detection. Current solutions rely either on highly localized data or on training with large quantities of background data. While efficient, the former may tend to over-segment. While more stable, the latter are less efficient and need adaptation to mis-matching data. Building on previous work in speaker recognition and diarization, this chapter reports a new binary key modelling approach to SCD which aims to strike a balance between efficiency and segmentation accuracy. The BK approach benefits from training using a controllable degree of contextual data, rather than relying on external background data, and is efficient in terms of computation and speaker discrimination. Parts of the work and analysis reported here were published in [173].

The chapter is organised as follows. Section 5.1 describes an introduction to the problem and related work. Section 5.2 discusses the role of the binary key background model (KBM) in the SCD process and its capacity as a context model. Section 5.3 elaborates on the methodology employed to perform SCD over BK-based segmental representations. Section 5.4 describes the experimental setup including databases, system configuration and evaluation metrics. Section 5.5 reports experimental results and discussion. A summary is presented in Section 5.6.

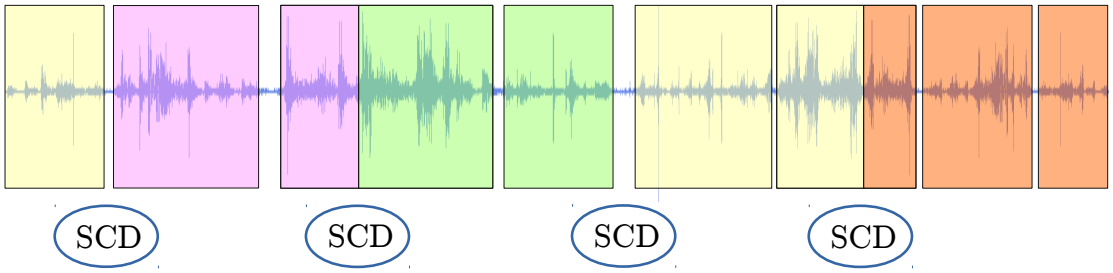


Figure 5.1: High level representation of the purpose of a SCD algorithm. Given an audio stream containing multiple speakers (each represented in a different colour), and after applying a voice activity detector, it will attempt to detect the boundaries between homogeneous speaker segments.

5.1 Introduction and related work

Speaker change detection (SCD), illustrated in Figure 5.1, also known as speaker turn detection and, more simply, speaker segmentation, aims to segment an audio stream into speaker-homogeneous segments (differently coloured segments in Fig. 5.1). Usually proceeded by a voice activity detector (VAD), SCD is often a critical pre-processing step or enabling technology; it is usually applied before other tasks such as speaker recognition or diarization.

The literature shows two general approaches. On the one hand, metric-based approaches aim to determine speaker-change points by computing distances between two adjacent, sliding windows. Peaks in the resulting distance curve are thresholded in order to identify speaker changes. The Bayesian information criterion (BIC) [46] and Gaussian divergence (GD) [174] are some of the most popular metric-based approaches. On the other hand, model-based approaches generally use off-line training using potentially large quantities of external data. An example of model-based approaches is the use of Gaussian mixture models (GMMs) [51], and universal background models (UBMs) [52]. More recent model-based techniques are based on the i-vector paradigm [53, 54] or deep learning (DL) [55, 56, 57, 175].

Despite significant research effort, SCD remains challenging, with high error rates being common, particularly for short speaker turns. Since they can operate upon only small quantities of data within the local context, metric-based approaches are more efficient and domain-independent, though they tend to produce a substantial number of false alarms. This over-segmentation stems from the intra-speaker variability in short speech segments. Model-based approaches, while more stable than purely metric-based approaches, depend on external training data and hence may not generalise well in the face of out-of-domain data.

The work reported in this chapter has sought to combine the merits of metric- and model-based approaches. The use of external data is avoided in order to promote domain-independence. Instead, the approach to SCD reported here uses variable quantities of contextual information for modelling, i.e. intervals of the audio recording itself. These intervals range from the whole recording to shorter intervals surrounding a hypothesized speaker change point.

The novelty of the approach lies in the use of an efficient and discriminative approach to context modelling binary keys (BKs) and cumulative vectors (CVs). The use of BK modelling for SCD is of interest in the general objectives to enhancements to the BK-based SD pipeline. In all recent work in BK-based SD by other authors [11, 149] and ours [155], segmentation consists in a straightforward partition of the audio stream into what are probably non-heterogeneous speaker segments. In this case, speaker segmentation is only done implicitly at best; none of the past work has investigated the discriminability of the BK approach for the task of explicit SCD.

The novel contribution of the work presented in this chapter includes two BK-based approaches to explicit SCD, which differ in the methods to the composition of the binary key background model (KBM). The proposed methods to compose the KBM support the flexible use of contextual acoustic information and are both compared to a classic metric-based approach that uses the Bayesian information criterion (BIC). They are also compared to DL based state-of-the-art approaches to SCD. Finally, the impact of SCD in the context of our SD pipeline is also assessed.

5.2 The KBM as a context model

In spite of their underlying differences, both metric- and model-based algorithms usually follow similar processing steps to perform SCD. A sliding window is used to process an audio stream and detect speaker changes within its content. Both approaches utilize exclusively the content within that sliding window. In this sense they are likely to be suboptimal: they neglect the acoustic information present in the remainder of the test session being analysed, i.e. the *context* that surrounds the sliding window. BK speaker modelling is characterised by the use of a KBM that captures the acoustic variability in the in-session data so that it remains independent from external training data. This context-modelling capacity of the KBM motivates our investigation of the use of BK speaker modelling as a solution to SCD. By means of the KBM composition, BK modelling allows SCD decisions to be made in a local sense, in a similar fashion to metric-based approaches, while considering variable amounts of the contextual in-session data, similar to model-based methods, in so that it leverages data not contained within

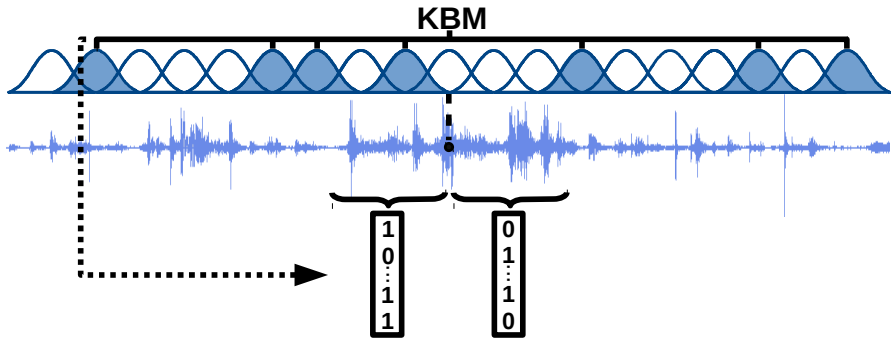


Figure 5.2: Global-context KBM obtained through the selection of Gaussians from a global pool.

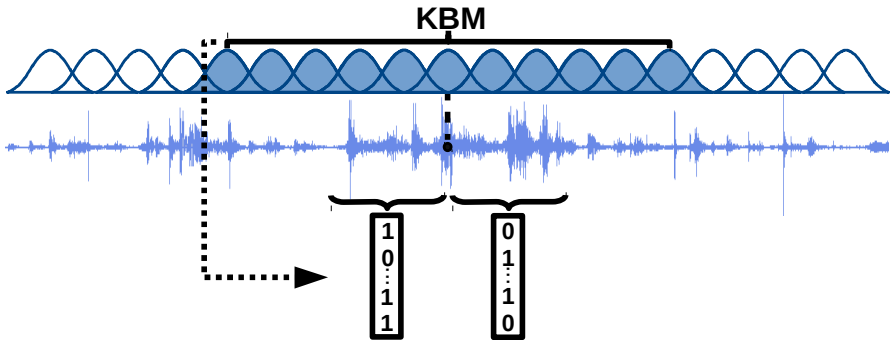


Figure 5.3: Local-context KBM constructed using all Gaussians estimated from within a local context.

the sliding window.

The extent to which the KBM is composed to cover the acoustic space in a meaningful way is the fundamental factor that determines the speaker discriminative capacity of BK speaker modelling. As introduced in Section 3.2.2, the KBM is composed by a selection of Gaussian components fitted to the test data itself. Once composed, the KBM represents the acoustic context of an audio file, and enables the estimation of segmental-level representations in the form of cumulative vectors (CVs) or binary keys (BKs).

In this chapter we explore two fundamental factors of importance to the usability of BK-modelling to SCD. First, the extent to which variable amounts of in-session contextual speech data, leveraged by means of a KBM, is beneficial to the task of SCD when comparing adjacent CVs or BKs; second, alternatives to KBM composition that are introduced in the next section.

5.2.1 KBM composition methods

We propose to evaluate the effect of the KBM as an agent for context modelling in SCD by means of two different composition methods. They are illustrated in Figures 5.2 and 5.3, and described in the following:

- ***Global-context* KBM:** an approach whereby the KBM is learned with data from the entire test sequence (Fig. 5.2). This approach follows the baseline algorithm described in Section 3.2.2 to choose the most discriminative and least redundant Gaussian components from within the Gaussian pool. Using a global-context approach to model the acoustic space allows the KBM, and hence the segment level CVs/BKs, to leverage acoustic cues that may lie far away from a potential speaker change point in a temporal sense but may be relevant from a speaker-discriminative perspective.

- ***Local-context* KBM:** an approach whereby the KBM is alternatively learned from a shorter context window centred on the hypothesised speaker change point (Fig. 5.3). Unlike the global-context approach, the local-context approach uses all the Gaussians contained in the defined context (no selection process is performed). This approach to KBM learning enables the flexible use of the acoustic context information that surrounds the hypothesised speaker change point.

5.3 BK-based speaker change detection

This section describes the use of CVs or BKs in SCD and provides a visual example of the speaker discriminative capacity of BKs.

SCD is performed using data from two non-overlapping windows, one either side of hypothesized speaker change points. CVs or BKs are extracted for each window and are compared using the cosine (Eq. 3.8) or the Jaccard distance (Eq. 3.10). This procedure is applied sequentially to obtain a curve of window distances at regular intervals. An example is illustrated in Figure 5.4 for a 2s speech segment. Local peaks in the curve represent speaker change candidates. Speaker change decisions are then obtained by thresholding the distance curve using an empirically optimised threshold θ.

By way of illustrating the speaker-discriminability of the BK approach, Figure 5.5 depicts a sequence of BKs extracted from an arbitrary speech fragment in the order of 2.5 minutes duration. Each column of the matrix is a BK computed from a 1s window with a 0.1s shift using a KBM of size $N = 320$. Speaker labels towards the top of the plot indicate the speaker which is active during each apparent segment. The vertical axis

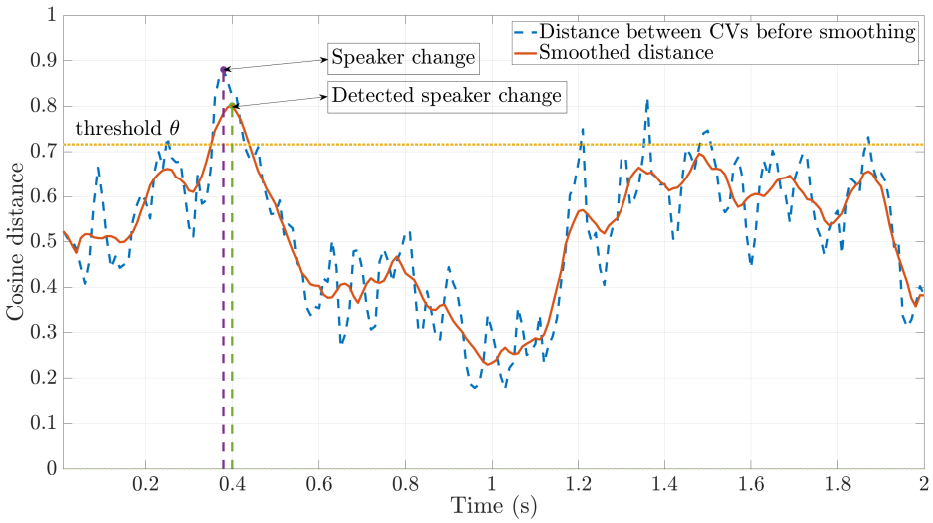


Figure 5.4: Speaker change detection process applied over an audio segment of 2s of speech from the ETAPE dataset by comparing adjacent CVs using the cosine distance. The resulting distance curve is smoothed to minimize the effect of outliers. Peak values of the smoothed curve that overpass the detection threshold θ are detected as speaker change points.

indicates the sorted KBM Gaussian indexes whereas the horizontal axis indicates time. The intra-speaker consistency of BKs is immediately evident, as are the inter-speaker differences which indicate speaker changes or turns.

5.4 Experimental setup

Experiments were designed to assess the efficacy of BK-based approaches to SCD and to compare the two KBM composition methods. This section describes the chosen database in Section 5.4.1. The configuration of baseline and BK-based approaches to SCD is described in Sections 5.4.2 and 5.4.3 respectively. The evaluation metrics are detailed in Section 5.4.4.

5.4.1 Database

The work reported here was performed with the ETAPE database [176] which contains audio recordings of a set of French TV and radio shows. It is composed of 3 partitions (18h for training, 5.5h for development and 5.5h for test). Temporal speaker identity annotations are provided only on 2 subsets of the training and development sets (with 18 out of 61 overlapping speakers). These annotations were generated in two steps. First,

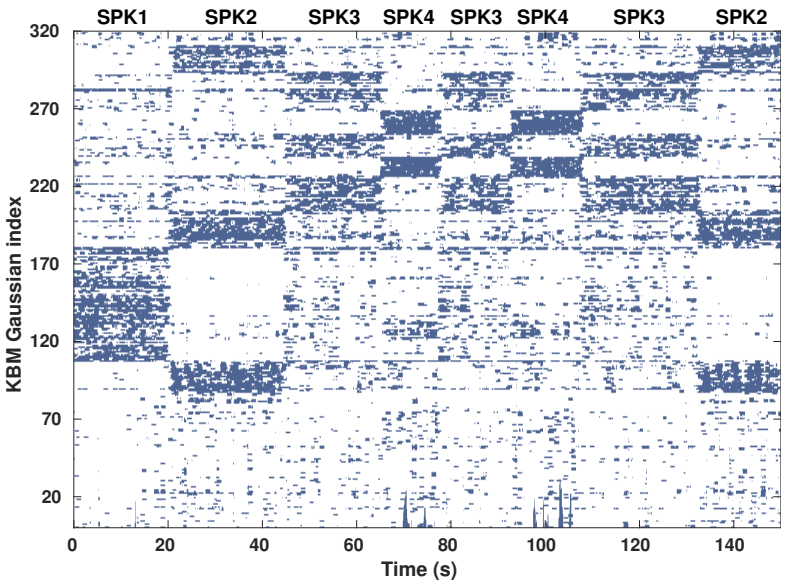


Figure 5.5: A matrix of BKs from an arbitrary 2.5-minute speech fragment from the ETAPE database. Each column of the matrix is an individual BK with N=320 elements extracted according to the procedure illustrated in Figure 3.5 and described in Section 3.4. Distinguishable BK patterns indicate distinct speakers whereas abrupt differences along the temporal domain indicate speaker change points.

automatic forced alignment was applied to the manual speech transcriptions upon which, in a second stage, trained phoneticians manually adjusted the boundaries. Annotations for the test set are coarser and not usable for the SCD task as, reported in the works in the literature to which we wanted to compare our results [55, 57, 59]. In keeping up with these results, the development set of the TV subset of the database was used for the experiments reported here.

5.4.2 Baseline SCD system

Acoustic features comprise 19 static Mel-frequency cepstral coefficients (MFCCs) which are extracted from pre-emphasised audio signals using an analysis window of 25ms with a time shift of 10ms using a 20-channel Mel-scaled filterbank. No dynamic features are used.

The baseline SCD approach is a standard BIC segmentation algorithm [46]. It is applied with two windows of 1s duration either side of a hypothesised speaker change point. The resulting BIC distance curve is smoothed by replacing each point with the average estimated over a 1s context. Local maxima are identified by enforcing a minimum distance of 0.5s between consecutive peaks. Within any 0.5s interval, only the highest peak is retained before speaker change points are selected by thresholding. This is a

standard approach similar to those reported in [177, 178, 179].

5.4.3 Binary key SCD system

Acoustic features are the same as for the baseline SCD system. Candidate Gaussians for the KBM pool are learned from windows of 2s duration with a time shift of 1s. The number of components in the final KBM is chosen adaptively according to a percentage α of the number in the initial pool. Reported below are a set of experiments used to optimise α. The number of top Gaussians M used for CV/BK extraction is set to 5 and the number of bits K that are set to 1 is set to 20% of the number of KBM components N.

Two BKs are extracted every 0.1s with sliding windows of 1s duration positioned either side of the hypothesized change point. The distance between each pair of CVs/BKs is calculated using the cosine/Jaccard similarity, and the distance curve is smoothed in the same way as for the baseline system. Speaker change points are again selected by thresholding.

5.4.4 Evaluation metrics

SCD performance is evaluated using the approach used in [57], namely through estimates of segment coverage and purity. Coverage is defined as:

$$\text{coverage}(\mathcal{R}, \mathcal{H}) = \frac{\sum_{r \in \mathcal{R}} \max_{h \in \mathcal{H}} |r \cap h|}{\sum_{r \in \mathcal{R}} |r|} \tag{5.1}$$

where $|r|$ is the duration of segment r within the set of reference segments $\mathcal{R}$, and where $r \cap h$ is the intersection of segments r and segments h within the set of hypothesis segments $\mathcal{H}$. Purity is analogously defined with $\mathcal{R}$ and $\mathcal{H}$ in Eq. 5.1 being interchanged.

An over-segmented hypothesis (too many speaker changes) implies a high segment purity at the expense of low coverage (hypothesised segments *cover* a low percentage of reference segments). In contrast, an under-segmented hypothesis (too few speaker changes) implies the opposite, namely high coverage, but low purity. Purity and coverage are hence a classical trade-off, with the optimal algorithm configuration depending on the subsequent task.

In order to concentrate on the assessment of SCD alone, ground-truth annotations are used for voice activity detection (VAD). It is noted that the use of ground-truth VAD as a hypothesis with a single speaker delivers a ceiling coverage of 100% and a floor purity of 83%. These values can be taken as a performance reference.

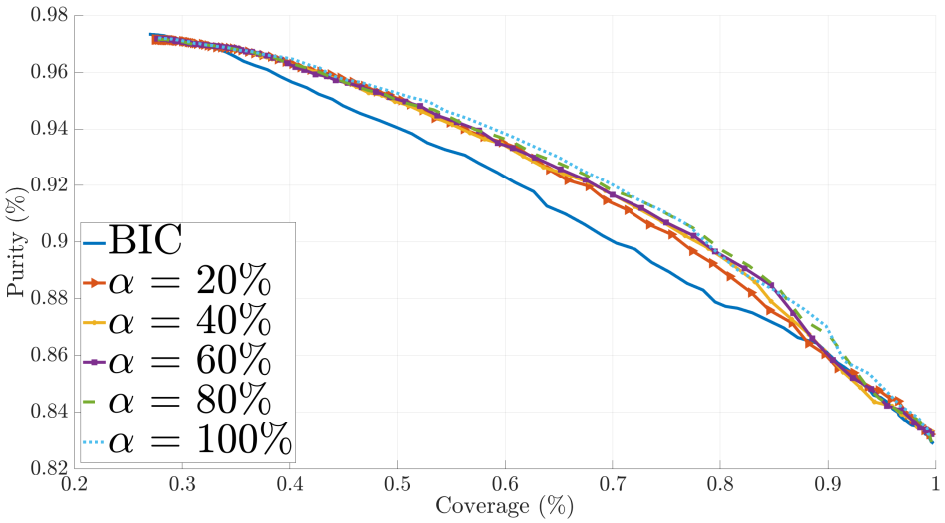


Figure 5.6: SCD performance measured in terms of segment purity and coverage using CVs and a global-context KBM, obtained by varying the decision threshold θ. Profiles are shown for different values of the KBM size (α).

5.5 Results

In this section, experimental results are presented for BK-based SCD using the two possible segmental representations derived from the technique, namely CVs (Section 5.5.1) and BKs (Section 5.5.2). These are extracted using the two proposed KBM composition methods. A comparison between the two variants is presented in Section 5.5.3. Finally, results that evaluate the impact of SCD upon a BK-based SD pipeline are presented in Section 5.5.4.

5.5.1 SCD using cumulative vectors

Figure 5.6 and Figure 5.7 show plots of purity and coverage for global- and local-context KBMs, respectively. Each profile shows the trade-off between the two metrics as the distance threshold θ is varied. Profiles are shown for KBMs whose size α is set to 20, 40, 60, 80 and 100 of the total number of original Gaussians. In both cases, the performance of the BIC baseline system is illustrated with a solid blue line.

Using CVs and a global-context KBM approach (Fig. 5.6), both purity and coverage increase for all sizes of the KBM over that of the baseline. While large values of α, which determines the percentage of final Gaussians selected to compose the KBM, give better performance as it increases, there is little improvement above $\alpha = 60\%$. Despite this saturation, the maximum performance gain is achieved for $\alpha = 90\%$. The trends is

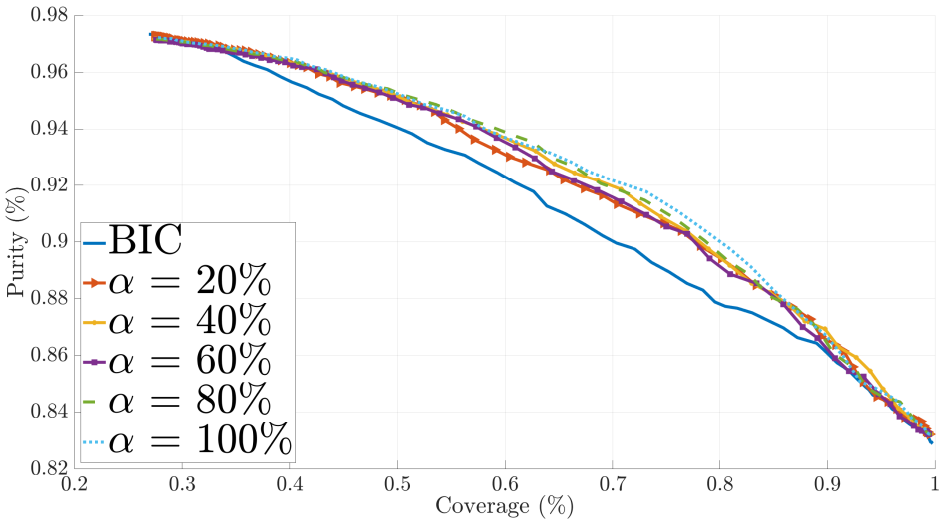


Figure 5.7: SCD performance measured in terms of segment purity and coverage using CVs and a local-context KBM, obtained by varying the decision threshold θ. Profiles are shown for different values of the KBM size (α).

similar when using CVs extracted from a local-context KBM (Fig. 5.7). Regardless of the small difference in performance for different values of α, differences with regard to the baseline are more significant as the KBM size increases. Although gains are somewhat inconsistent as compared to the global-KBM approach, they yield a slightly bigger overall increase in performance.

An alternative visualization of the results is presented in Figures 5.10 and 5.11, which plot the average relative increase in segment coverage and purity, respectively, over that of the baseline system. In the case of CV-based approaches to SCD (solid blue line for global-context KBM and arrow-marked, orange line for local-context KBM in Fig. 5.10), the average relative increase in coverage increases from around 6% for smaller KBMs ($\alpha \sim 20\%$) to nearly 10% for KBMs that cover the complete acoustic space ($\alpha = 100\%$). However, average relative increases in purity values (Fig. 5.11, same colors and markers) are more modest than those in coverage. Global- and local-context based approaches to KBM composition using CVs benefit purity in a similar tone. For small sizes of KBM the average relative purity improvement obtained is in the order of a 2%. Alternatively when the KBM is allowed to accumulate larger amounts of context, the local-context KBM marginally outperforms the global-context KBM.

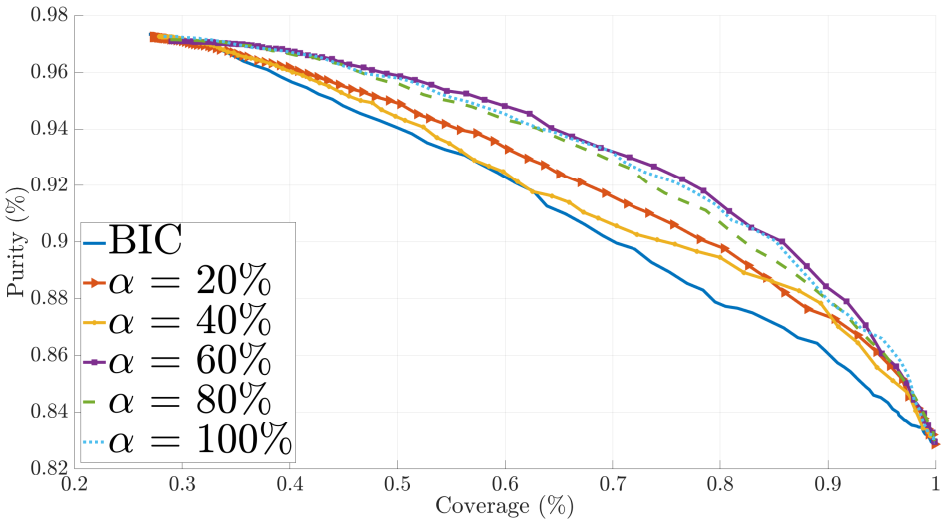


Figure 5.8: SCD performance measured in terms of segment purity and coverage using BKs and a global-context KBM, obtained by varying the decision threshold θ. Profiles are shown for different values of the KBM size (α).

5.5.2 SCD using binary keys

In similar fashion to results presented in Section 5.5.1, Figure 5.8 and Figure 5.9 represent purity and coverage for global- and local-context based KBMs, respectively, but now using BKs instead of CVs. The BK approach with global-context KBMs (Figure 5.8) gives universally better performance than the baseline, even if the trend is somewhat inconsistent. Larger KBMs then give better performance, e.g. for α greater than 40%. The optimal α is 60%. Larger values of α do not necessarily give better performance. The BK approach with local-context KBMs (Figure 5.9) also outperforms the baseline. While the trend is consistent for lower values of coverage, across the range the optimal α varies between 60% and 100%.

The results presented in Figure 5.10 and Figure 5.11 provide a clear visualization of the gains when using BKs. A global-context KBM gives an average relative increase in segment coverage of 17.4% for $\alpha = 60\%$, while a local-context KBM brings the improvement to 18.3%. Similarly, maximum gains in average relative purity of 4.5% are achieved for KBM sizes of $\alpha = 60\%$, while local-context KBM derived BKs achieve a more consistent performance gain at α values ranging between 70% and 90%.

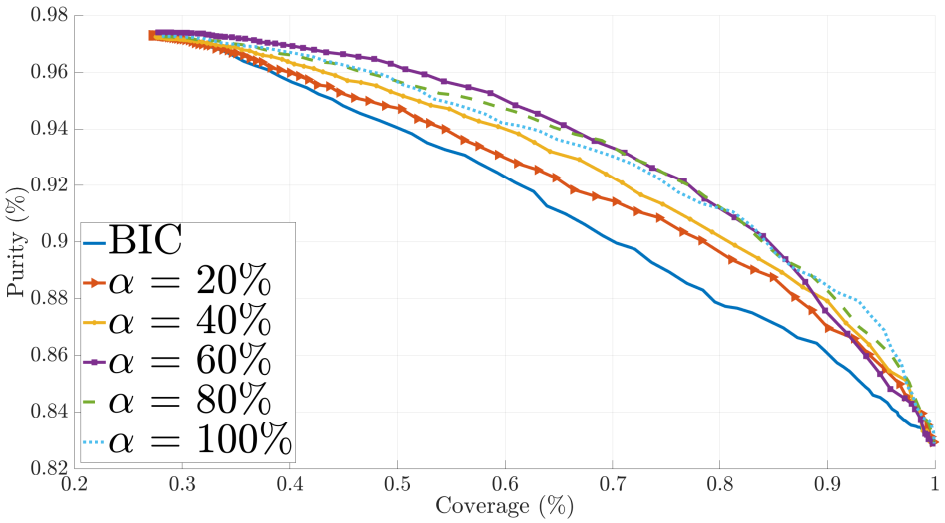


Figure 5.9: SCD performance measured in terms of segment purity and coverage using BKs and a local-context KBM, obtained by varying the decision threshold θ. Profiles are shown for different values of the KBM size (α).

5.5.3 Comparison between BK-based SCD systems

Results for both CV- and BK-based SCD exhibit significant differences. CVs and BKs were tested in different KBM composition conditions and with different KBM sizes. Nonetheless, the extraction of the segmental representations remained independent of the KBM composition method. That is, for a KBM size of $\alpha = 70\%$ using a global-context approach, a unique KBM is employed to derive CVs and BKs. The same same approach is followed when using local-context KBMs. Performance differences between CVs and BKs, despite the use of identical KBMs for both BK & CV representations, give rise to some intriguing questions.

Why do CVs not benefit from larger KBM sizes like BKs? Figure 5.10 and Figure 5.11 show that the gain derived from increasing KBM sizes saturates much sooner, and in a less significant manner, for CVs than BKs. BK behavior w.r.t the KBM size is, on the one hand, intuitive when considering the KBM composition mechanism: larger KBMs can leverage more contextual information. The extent to which this is beneficial is limited to the amount of non-redundant information available in an acoustic space that the KBM composition method manages to capture, establishing an upper bound to the KBM size in its benefit to performance. Larger KBM sizes are likely to capture redundant acoustic content and degrade performance. Why, then, does this not apply also to CVs? The difference in performance may stem from two factors: one is the difference in the process of CV and BK composition; the other is the distance measures

Purity (%)			84	88	92	96
Coverage (%)	**BIC**		96.48	79.54	60.92	37.90
	CVs	**Global KBM** ($\alpha = 100\%$)	98.46	87.80	69.75	43.91
		Local KBM ($\alpha = 90\%$)	98.04	85.90	70.78	44.25
	BKs	**Global KBM** ($\alpha = 60\%$)	98.88	91.71	**78.46**	48.99
		Local KBM ($\alpha = 70\%$)	**98.99**	**92.86**	77.45	**51.51**

Table 5.1: Coverage obtained by employing global- and local-context KBM composition methods for CVs and BKs in the task of SCD. KBM size (α)is chosen for each system to maximize the gain in coverage following that reported in Figure 5.10.

employed to detect speaker change points. On the one hand, the difference between the generation of CVs and BKs is a quantization step. To compose a BK, a given CV's top-K activated KBM elements are set to 1 while the remaining positions are set to 0. Results indicate that this quantization step allows BKs to be more discriminative than CVs, thereby resulting in better SCD performance. BKs are compared by means of a Jaccard distance, as per the cosine distance of the CVs. Whilst the cosine distance emphasises the angular difference between two CVs, the Jaccard distance (see Section 3.5) performs a more aggressive distance measurement (similar to a sort quantization) whereby only very different BKs result in high distances. It is important to remember that CVs and BKs are extracted over speech segments that are extremely short, i.e. 1s long. In addition, the sliding window of 0.1s employed also implies the speech content in adjacent windows is very similar. Such a short amount of speech and high overlapping may easily derive in noisy speaker representations. The use of bolder means of quantization in both the segment extraction (that of BKs over CVs) and distance scoring (that of Jaccard vs. cosine distance) may thus be fundamental to the final BK-based SCD performance.

What is the best approach to use the KBM as a context model for SCD? While performance using BKs clearly outperforms the baseline and CVs in both coverage and purity (again, Fig. 5.10 and Fig. 5.11), the question remains as to which approach to KBM composition brings the most benefit when using BKs. It is of interest to compare the two proposed methods not only in terms of performance, where the differences are small, but also in terms of efficiency and practical application. Even if the local-context approach slightly outperforms the global-context method for BKs, each approach can be better suited for different applications. On one hand, in the case of offline processing (when the entire input stream is available in advance), the global-context approach is more efficient since the KBM is fixed for the entire process, hence frame-wise likelihoods can be computed only once and then reused for subsequent operations. However, in the local-context approach, the KBM changes over time (using Gaussian components

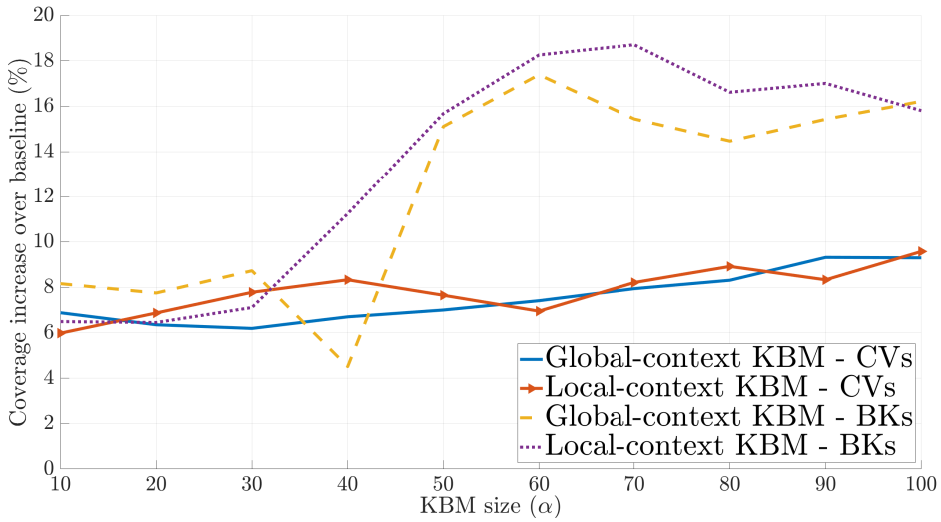


Figure 5.10: Average increase in segment coverage (%) for the different KBM composition methods and segment representations over that of the baseline.

estimated from the context window). This implies the recomputation of frame-wise likelihoods every time the window is shifted, therefore implying an extra computation cost. On the other hand, in online processing scenarios, the global-context approach cannot be used since the complete input stream is required in advance to train the KBM. However, the local-context approach is well suited to online applications since it utilises only local information. In the latter case, system latency would be proportional to the amount of contextual data considered.

Table 5.1 illustrates the variation in coverage against purity which could be of importance to specific post-SCD application, such as SD or recognition. The best performing systems are compared in terms of α and performance is compared to that obtained with the baseline system. The BK approach gives higher coverage at *all* operating points, especially for those with higher purity. This is important as it validates the viability of our approach to SCD independently of the optimization of the KBM size (defined as a percentage by means of α) as reported here. These improvements, which are similar to results achieved with DL-based solutions in the same dataset [57] and fall only slightly behind those of more advanced approaches [59], despite the fact that our approach does not use any external training data.

5.5.4 Speaker diarization using a BK-based SCD

A last experiment is considered to assess the impact of SCD variations upon the baseline SD pipeline as defined in Section 3.7. Different methods to speech segmentation, and

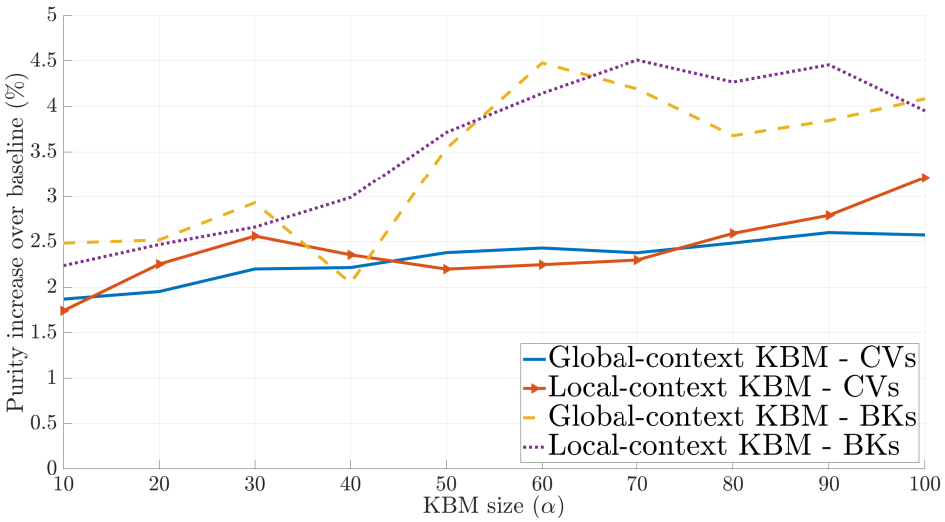


Figure 5.11: Average increase in segment purity (%) for the different KBM composition methods and segment representations over that of the baseline.

implicit and *explicit* SCD are considered. These approaches to SCD are illustrated in Figure 5.12 (by means of their application upon an example speech signal in Fig. 5.12a) and are defined as follows:

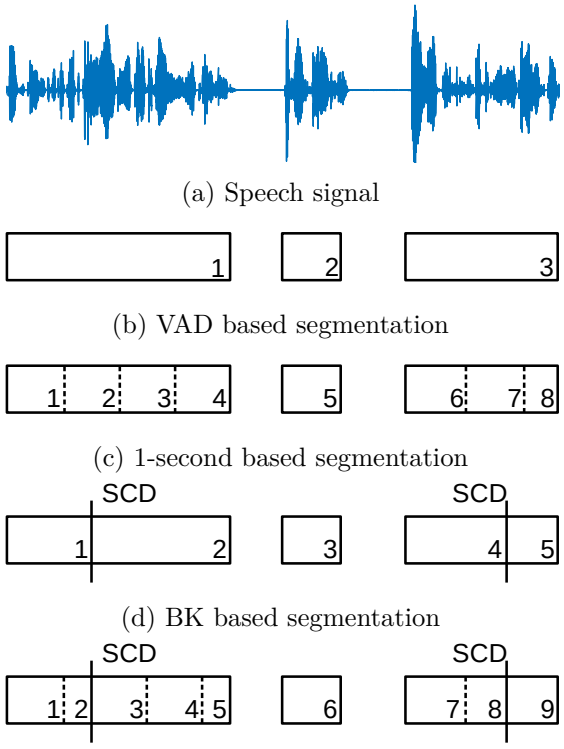
Voice activity detection (VAD): This method (Fig. 5.12b) does not perform any segmentation other than those between speech and non-speech segments.

1-second: Upon this method, homogeneous 1s-long speech segments are applied on top of the VAD-derived segmentation (Fig. 5.12c). This is the speech segmentation approach defined in Section 3.7 for the baseline BK-based SD system.

Binary key (BK): Here, a binary key-based SCD is used in the SD pipeline (Fig. 5.12d). It employs a local-context KBM with an optimal KBM size fixed to $\alpha = 70\%$, and with the threshold θ set to derive a segment purity of 94%. This purity value is considered in keeping with related literature [57, 59].

Combined: This approach to speaker segmentation merges the 1-second segmentation and the BK-based split (Fig. 5.12e), in an attempt to leverage both methods at the same time.

Results are presented in Table 5.2 in terms of the diarization error rate (DER, in % using a standard forgiveness collar of 0.25s). Following our findings reported in Chapter 4 [155], we explore the use of MFCCs using frame lengths of different sizes, specifically 25 and 128ms (alternative front-ends in terms of spectral analysis like those



(a) Speech signal

(b) VAD based segmentation

(c) 1-second based segmentation

(d) BK based segmentation

(e) Combined 1-second and BK segmentation

Figure 5.12: Illustrative example of different SCD approaches applied to the pipeline of the BK-based SD baseline.

of Chapter 4 were not tested here as they are considered beyond the scope of this SCD-influence analysis). Results are also presented with and without a GMM-based resegmentation.

The 2nd and 3rd rows of Table 5.2 illustrate performance obtained using MFCCs extracted from 25ms-long speech frames. A VAD-derived segmentation (4th column of Tab. 5.2) indicates a floor of performance of 31.95% and 27.9% for a system without and with final resegmentation, respectively. This approach delivers the worst performance, which is expected considering the completely speaker-independent segmentation generated by the VAD system. The baseline segmentation system based on 1-second splitting of the speech data (5th column of Table 5.2) gives performance closer to that reported in [55](21.12%) with DERs of 24.01% and 23.3% for systems with and without resegmentation. A homogeneous 1s splitting of the audio data seems sufficiently fine-grained to generate some sort of implicit speaker segmentation, thereby leading to improved performance. The performance of the proposed BK-based approach to SCD is illustrated in the 6th column of Table 5.2. It delivers DERs of 22.11% (8% relative improvement over a 1-s segmentation) and 20.01% (15% relative improvement over the 1-s baseline).

			VAD	1-second	BK	Combined
Frame length	**25ms**	**w/o reseg.**	31.95	24.01	**22.11**	25.2
		w/ reseg.	27.9	23.3	**20.01**	23.04
	128ms	**w/o reseg.**	33.49	**18.79**	22.52	21.37
		w/ reseg.	30.23	**17.84**	20.68	19.6

Table 5.2: Results in terms of DER for the different SCD methods applied to the BK-based SD pipeline. It is measured for different frame lengths used in feature extraction, and including/excluding the final resegmentation step.

These results verify the benefit of our approach to SCD upon a fully fledged SD system when using 25ms frame lengths. It is interesting to note how the difference between the two approaches is more substantial when resegmentation is applied than when it is not. Purer, longer segments are derived from a BK-based SCD approach which probably avoid speaker change points missed by the 1-second approach to segmentation. This allows for purer initialization of the GMM-based resegmentation algorithm. Results for the last method that combines 1-second segmentation and BK-based SCD are presented in the 7th column of Table 5.2. Ideally, this approach could exploit the best of both BK-based SCD and 1-second approaches to segmentation by implicitly and explicitly determining boundaries for speaker-homogeneous speech segments. However, results show performance does not seem to benefit, with 25.5% and 23.04% DERs for systems with and without resegmentation, both similar to that of the 1-second segmentation. These results suggest that the 1-second long segmentation does not benefit from additional SCD-derived boundaries, as they probably generate excessively short segments. It can also be argued that a purely BK-based SCD suffers from the over-segmentation used in a 1-second approach, as it effectively dilutes the purpose sought by the BK-based SCD in terms of maximising segment coverage for a given purity.

Following, rows 4th and 5th of Table 5.2 present results obtained when using MFCCs extracted using frame lengths of 128ms. Results in Chapter 4 highlight the benefit of using longer frame lengths for feature extraction for BK-based SD, and are consequently used here for the sakes of verification. Performance using a VAD approach decreases with regard to the result obtained using a 25ms configuration, reaching DERs of 33.49% and 30.32%, with and without resegmentation respectively. Under this feature extraction configuration, a 1-second segmentation delivers a substantial increase in performance, with DERs of 18.79% and 17.84%. These results are in contrast to the trend observed in the 25ms frame feature extraction configuration. By using a 128ms frame length the BK-based segmentation does not deliver any improvement w.r.t the 1-second segmentation, performing at a similar level than that of the 25ms configuration (22.52% and 20.68%).

The combination of both techniques does not deliver any improvement either, resulting in DERS of 21.37% and 19.6%. The results derived from using a 1-second segmentation approach and 128ms frame lengths verify the impact of frame length for BK-based SD performance. This impact is so important as to outperform the possible benefit of purer, longer speaker-homogeneous speech segments. As to the reason behind it, results indicate that the coupling of a 1-second segmentation with 128ms frame feature extraction benefits the feature binarization and segment extraction process (see, respectively, Sections 3.3 and 3.4), generating more speaker-discriminative representations.

Results presented in this section show the benefit of the BK-based approach to SCD over a 1-second homogeneous segmentation when a 25ms frame length is employed in the extraction of MFCCs. However, for frame lengths of 128ms the tendency is reversed and a system independent of explicit SCD, such as that of a 1-second segmentation, obtains better results. This finding corroborates those reported in literature [55, 59]. Future work should explore a better leveraging of the explicit SCD methods that could be complementary to the 1-second segmentation.

5.6 Summary

This chapter introduces a binary key (BK) solution to speaker change detection (SCD). The algorithm uses traditional acoustic features and a configurable quantity of contextual information captured through a binary key background model (KBM). Speaker-discriminative CVs and BKs are then extracted from the comparison of acoustic features to the KBM. The binarization of acoustic features resembles a form of quantization which helps to reduce noise and hence improve the robustness of subsequent SCD. The latter is performed by thresholding the distance between CVs/BKs extracted from two adjacent windows either side of hypothesized speaker change points. While not requiring the use of external data, two variants of KBM composition are shown to outperform a baseline approach based on the classical BIC. Results obtained using a standard dataset show average relative improvements which compare favourably to results reported recently for more computationally demanding solutions based on DL. The impact of the explicit SCD process is also measured in terms of benefits to the SD solution reported in this thesis. Results highlight the challenge of the task to better integrate SCD decisions in the BK-based diarization pipeline which could be explored in future research.

Chapter 6

Leveraging spectral clustering for training-independent speaker diarization

The first DIHARD challenge aimed to promote speaker diarization (SD) research and to foster progress in domain robustness, by proposing challenging SD scenarios that are new to the research community. This chapter reports EURECOM's submission to the DIHARD challenge and the work undertaken to tackle the challenging dataset. This work was published in [180]. EURECOM's DIHARD submission was based upon our low-resource, domain-robust binary key approach to speaker modelling. Contributions include the first application of spectral clustering (SC) to BK-derived cumulative vectors (CVs) as an alternative to agglomerative hierarchical clustering (AHC), as well as its use for estimating the number of speakers and a mechanism to detect single-speaker trials. Experimental results obtained using the standard DIHARD database show that the contributions reported in this chapter deliver relative improvements of 39% in terms of the diarization error rate over the baseline algorithm. An absolute DER of 29% on the evaluation set compares favourably with those of competing systems, especially given that the binary key system is highly efficient, running 63 times faster than real-time.

The remainder of this chapter is organised as follows. Context and motivations to the work are given in Section 6.1. The difficulties steaming from domain variability to SD, as well as the implications of domain variability as concerns the DIHARD challenge, together with a brief description of the dataset, are given in Section 6.2. Section 6.3 describes preliminary work done to assess the limitations of our baseline system that motivated

the contributions reported in this chapter. Section 6.4 describes the investigation of SC and its application to a SD system based on BK speaker modelling. Experiments and results are described, respectively, in Sections 6.5 and 6.6. A summary of the work and findings reported in this chapter is given in Section 6.7.

6.1 Context and motivation

While SD research attracted significant interest in the past, the field has somewhat stagnated in recent times. This is perhaps due to the lack of significant datasets that explore domains beyond certain categories. For meeting-based conversation, the once popular NIST Rich Transcription evaluations [181] is now seemingly discontinued. More recent databases such as those used for the ETAPE [176], REPERE [182], AMI [183], or the Albayzin [12, 13] evaluations are either modest in size and/or put their focus exclusively on specific domains, e.g. broadcast news, meetings or televised chat shows. As a result, each database and evaluation has a somewhat limited audience.

The DIHARD initiative [14] was born to re-energise the research effort. The availability of a large dataset supporting a broader range of application scenarios, e.g. including medical interviews, conversations involving children, even monologues, rejuvenated research interest fostered progress in domain-robust SD; the DIHARD dataset contains no training data and represents the broadest domain variation captured in a single SD dataset to date.

There are two distinct approaches to address such a challenge. The first entails the optimisation of systems using a large quantity of training data that spans adequately the domain variation captured in the DIHARD data. The second is an inherently domain-neutral approach that requires no background training data, or rather acquires background data from acoustic streams at runtime. A hybrid approach might aim to exploit the benefit of background training data, but with the facility to adapt to a specific domain at runtime.

Given our interest in this thesis in low-resource and computationally efficient, practicable SD technology, our efforts to address the first DIHARD challenge explored the second approach. Past work showed the merit of the binary key approach to SD [11, 155] that does not require any background training data. Thanks to this independence from training data, it is ideally suited to domain-robust diarization. However, while its principal merit relates to computational efficiency, rather than raw performance, it is not necessarily expected to be competitive with the best-performing submissions to the first DIHARD challenge. The difficulty of the proposed domains motivated the exploration of improvements to our SD pipeline that do not imply learning from external data and can

be tuned within a development set. Results show nonetheless that, with the introduction of three modifications, it remains competitive with even the best submissions, while still offering advantages in terms of computational efficiency.

Explored improvements that also have no need for external training involved in the work reported in this chapter include front-end processing, alternatives to AHC clustering, approaches to estimate the number of speakers, and a dedicated mechanism to detect single-speaker trials. For the front-end, infinite impulse response - constant Q Mel-frequency cepstral coefficients (ICMCs), whose use upon BK speaker modelling is analysed in Chapter 4 [155], were also tested using this new dataset. Clustering alternatives to AHC as well as clustering selection improvements are based upon SC. A established method in its application to SD [8, 119, 120], SC is applied here for the first time to a SD system based on BK modelling. Finally, and also based on a method derived from SC, we investigated an approach to single-speaker detection as a means of overcoming significant impact upon performance introduced by the under-clustering of single-speaker trials.

6.2 The first DIHARD challenge

The performance of state-of-the-art SD solutions is reasonably high in some domains. One is telephony, where clustering is applied to DNN-based embeddings (see Chapter 2), e.g. using datasets such as the NIST-based CALLHOME dataset [184], formed of conversational telephone speech [7, 8, 185]. Alternatives, such as approaches based upon Variational Bayesian inference have achieved similar results [116]. At the time of writing, state-of-the-art results within the telephony domain falls in the 8-10% DER range. The relatively good performance here may be associated to a number of factors. To name a few, telephony conversation are rather structured with regard to speaker turns, while participants in the conversation speak near to their respective microphones. Also, the number of speakers is commonly limited to 2, and the amount of overlapping speech is often relatively small.

A different scenario in which research in SD has been active, but where performance is not so high, is that of broadcast radio and TV content. There have been multiple evaluations in this domain over the years, such as the ETAPE [176] and REPERE [186] campaigns mentioned above, or the more recent Albayzin evaluations [12, 13]. Work reported in the literature that uses DNN-based embeddings [6] as well as i-vectors [170], or our approach to diarization based on BK speaker modelling [11, 155] have achieved performances that typically range between 15-20% DER. Despite the usually controlled conversational scenario of broadcast content, diarization performance in this domain is

usually worse, with a number of factors being responsible. Among them, there is usually greater variation in the number of speakers per session, and there are often other acoustic sources, i.e. music and noises.

Another domain is that of speech recorded in meeting rooms. Relevant datasets include the NIST Rich Transcript (RT) evaluations [187, 188, 189], or the Augmented Meeting Interaction (AMI) dataset [183]. When SD is performed using speech data from a single-distant microphone (SDM), results reported in the literature typical DERs lie in the 20-25% [190, 191] range. The meeting domain is characterised by a varying, unknown number of speakers, highly unstructured conversations, and poorer recording conditions, making of it among the most challenging use cases for SD.

The above discussion is testament to the impact of domain variability (in the form of unstructured speech content, additional noises and nuisances, and variable number of speakers) upon the performance of today's SD systems. When the domain is known a priori, then SD systems can be generally tuned to be reliable. However, as variability increases performance decreases. The DIHARD challenge was the first significant initiative to promote the study of domain robustness in SD.

The dataset provided in conjunction with the first DIHARD challenge [14] is a composition of different data subsets. Together, they represent domains in which the application of SD could be of benefit as a pre-processing technique, i.e. as a precursor for speech or speaker recognition, including domains that have not been broadly explored previously through the work & campaigns described above. A brief description of the different data subsets is given below:

- **Child language acquisition recordings (SEEDLINGS)**: Recordings from the Study of Environmental Effects on Developing Linguistic Skills (SEEDLingS) dataset. These include home-recordings involving children between 6 to 18 months of age that are learning to speak and people interacting with them.
- **Supreme Court oral arguments (SCOTUS)**: Oral arguments from the 2001 term of the U.S. Supreme Court. Channels recorded from table-mounted microphones were summed and recorded in a single channel.
- **Map tasks (DCIEM)**: Recordings of subjects involved in map tasks gathered from the DCIEM Map Task Corpus. In this scenario, a 'leader' sits across the table from a 'follower'. The latter must follow oral instructions of the leader to find his path across a paper map.
- **Clinical interviews (ADOS)**: Recordings of Autism Diagnostic Observation Schedule (ADOS) interviews organised at the Center for Autism Research at the

Children's Hospital of Philadelphia in the United States. Audio was collected from a video camara mounted on a wall nearly 4 meters away from the interview.

- **Radio interviews (YP)**: Interviews made in a student-run radio program of the late 1970s, YouthPoint (YP) at the University of Pennsylvania in the United States.

- **Sociolinguistic interviews (SLX)**: Field recordings conducted during the 1960s and 1970s across the Americas and the United Kingdom belonging to the SLX Corpus of Classic Sociolinguistic Interviews.

- **Meeting speech (RT04S)**: Recordings of multi-party meetings collected from the 2004 Spring NIST Rich Transcription (RT-04S) dev and eval partitions. These were recorded at multiple locations with a different microphone setup. For DIHARD, a single channel was distributed for each meeting, corresponding to the RT-04S single distant microphone (SDM) condition.

- **Audiobooks (LIBRIVOX)**: A rather unsual scenario for SD where single-speaker, amateur recordings of audiobooks selected from LibriVox are provided.

- **YouTube videos (VAST)**: Content from online videos collected from the Video Annotation for Speech Technologies (VAST) project. The recording conditions, thematics and languages (English and Mandarin) are heterogeneous within the partition.

6.3 An analysis of our baseline

Given the diverse conditions in the DIHARD dataset, a first analysis of our system was necessary to determine our baseline performance and our optimal performance goal by means of controlled experiments. This section describes the preliminary work done using our SD pipeline based on BK speaker modelling. In Section 6.3.1 our baseline system is briefly defined. Section 6.3.2 reports experiments and results using the baseline system. Finally, the weaknesses identified in the system that provided scope for research and the contributions reported in this chapter are discussed in Section 6.3.3.

6.3.1 The baseline system

The pipeline of the baseline system for the DIHARD challenge is that reported in Section 3.7. Baseline configuration details were borrowed from our previous and successful participation in the Albayzin 2016 challenge reported in Chapter 4. The following provides a brief reminder following the pipeline illustrated in Figure 3.7 of Chapter 3:

- **Features:** Acoustic features (MFCCs) are extracted from the raw audio signal by means of a sliding window.

- **Voice activity detection:** Oracle annotations were used to remove silence from the audio session.

- **Segmentation:** Remaining speech is split in overlapping segments of 3s of length with a 1s shift between segments.

- **KBM training:** A pool of G session-dependent Gaussian components is learned from the in-session acoustic content, and a KBM is derived using the discriminative process described in Section 3.2.2. The dimension of the KBM N is determined as a percentage α of the Gaussian pool size G. Following our work reported in Chapter 4, $\alpha = 85\%$ was set used for the baseline system.

- **Segment/cluster representation:** Feature binarization as explained in Section 3.3 is applied in order to derive speaker-discriminative cumulative vectors (CVs) as described in Section 3.4.

- **Clustering:** This stage is composed of two parts. First, a bottom-up AHC algorithm is applied, which is illustrated in Fig. 3.8 and described in Section 3.7. Follows a clustering selection mechanism using an elbow criterion linked to the within-class sum of squares (WCSS) of the AHC-derived solutions [149].

- **Resegmentation:** A final resegmentation step based on GMMs and frame-level reassignment is applied to refine the hypothesized segment boundaries.

Leveraging recent front-end improvements

In work reported in Chapter 4 of this thesis, front-end improvements (motivated by recent success in the application of multi-resolution spectral analysis for voice biometrics applications [160, 161]) are applied and measured for the first time to BK speaker modelling, in the form of Infinite Impulse Response-Constant Q (IIR-CQT) Mel-frequency cepstral coefficients (ICMC) [160]. Performance using CVs increases in terms of equal error rate (EER, measuring speaker recognition) and diarization error rate (DER, measuring SD) when tested using a dataset formed of TV content [155]. The promising results of our previous work encouraged incorporating ICMCs to our baseline system analysis, so that their benefit with regard to MFCCs could be measured in the context of diverse SD domains included in the first DIHARD challenge.

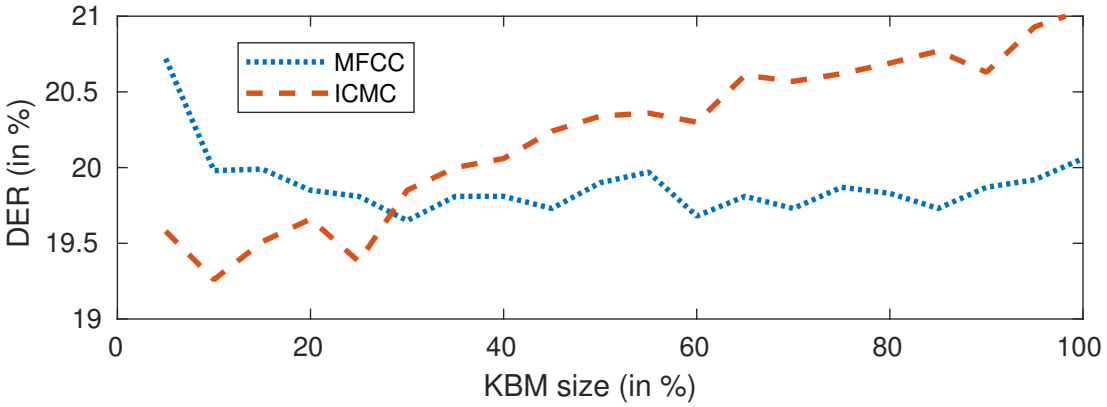


Figure 6.1: Comparison of MFCC and ICMC features for different KBM sizes when using an oracle selection of clustering solutions (those which minimise DER) on the development set of the DIHARD dataset.

6.3.2 Experiments and results

Baseline performance was measured by means of two different SD experiments on the development partition of the DIHARD dataset. Results were compared using different methods to determine the number of speakers and, at the same time, comparing MFCC and ICMC-based front-ends. First, the baseline system was applied *as is* following our Albayzin 2016 best configuration [155]. In the second experiment, DER was obtained by selecting, in an oracle manner, the clustering solution generated by our baseline AHC algorithm that maximizes performance on a per-session basis.

Elbow-based speaker estimation

In this experiment, the full baseline pipeline was applied to the DIHARD development set, using an elbow criterion [149] to determine the number of speakers. MFCCs achieved a DER of 44.5% while ICMCs reached a 44.8%. While performance was poor, the differences between front-ends were small and non-conclusive as to determine which features to use in our approach to the DIHARD dataset.

Oracle speaker estimation

Figure 6.1 illustrates the DER[1] profiles of our baseline system when the number of speakers is estimated in an oracle manner as a function of the KBM size α. Curves are plotted for MFCC and ICMC based front-ends. The best performance is achieved for MFCCs with a KBM fixed at $\alpha = 30\%$ with a DER of 19.7%. The tendency for

[1]Note DER is calculated without any forgiveness collar as was indicated for the DIHARD evaluation.

MFCC features is nonetheless almost constant, and similar levels of performance are achieved with both small and large KBM sizes. On the other hand, results based on ICMC features achieve better performance at 19.3% DER with a KBM of size $\alpha = 10\%$. The use of larger KBM sizes for ICMC features degrades the performance.

6.3.3 Identifying the baseline strengths & weaknesses

The comparison of the results for the two experiments reported above (automatic vs. oracle estimation of the number of speakers and front-end comparison) showed a small, even negligible difference between MFCCs and ICMCs when an automatic criterion is used to determine the number of speakers per session. Nonetheless, the oracle speaker estimation experiment showed ICMCs to provide a slight improvement over MFCCs. At the same time, the importance of the final size of the KBM is different for both front-ends. MFCC-based KBMs do not seem to be affected by bigger KBM sizes, suggesting the composition algorithm (Section 3.2.2) might not be working appropriately in this challenging dataset. However, the same KBM composition algorithm seems to work better for an ICMC-based front-end. Smaller KBMs sizes are capable of capturing more discriminative acoustic information when using ICMCs, hence benefiting performance. The small but positive difference in the oracle baseline performance, where the variability in the trend of the KBM size suggests the discriminative KBM composition process is working better for ICMCs than for MFCCs, and the knowledge gained from our work reported in Chapter 4, suggests ICMC features could potentially lead to better overall performance. The ICMC front-end was consequently chosen for our DIHARD work.

On a different line, the results obtained by selecting the number of speakers in an oracle manner brought error rates to DERs below 20%, which is less than half that of the fully automatic baseline. This large difference in results suggests that, while the applied AHC algorithm is capable of generating diarization hypotheses that offer much more reasonable performance, the clustering selection algorithm was consistently choosing a sub-optimal number of clusters. This evidence highlighted the importance of appropriately determining the right number of clusters as a possible path to improving the performance. Keeping into account the overall objective of the work reported in this part of this thesis, i.e. remaining independent of external training data, motivated the study of alternative methods to more reliable and domain-robust approaches to clustering and to the estimation of the number of clusters/speakers. In particular, the work reported in the remainder of this chapter illustrates efforts to adapt and leverage spectral clustering (SC) [192] in its first application to BK-based SD.

6.4 Spectral clustering

This section relates to the enhancements to BK-based SD explored using SC. An introduction to SC and its possible benefit to our work in the DIHARD challenge is given in Section 6.4.1. Its application to BK speaker modelling is detailed in Section 6.4.2. Last, its use as a single-speaker session detection module is proposed in Section 6.4.3.

6.4.1 Introduction and motivation

Clustering, i.e. the partitioning into dissimilar groups of similar items, is an extremely challenging problem that has been explored in literature for decades [193]. Common approaches to clustering can be divided into two branches: *partitional* and *hierarchical*. Most approaches to clustering in SD fall within the second category. Hierarchical clustering algorithms perform an iterative, nested, agglomerative (bottom-up) or divisive (top-down) process upon data points, e.g. segment/cluster level speaker representations, acoustic features, etc. in the case of SD. The reason why *hierarchical* clustering is more widely applied to SD than that of *partitional* clustering may relate to the easier estimation of the normally unknown number of speakers by clustering derived from the former rather than by the latter. The nested character of *hierarchical* approaches enables for the design of simpler mechanisms to stop the clustering process, thanks to the explicit relationship that exists between clusters merged/divided by successive iterations of the AHC algorithm. Such methods usually depend on the tuning of a threshold θ on development data to determine the stopping criterion of the AHC process, whereby the number of speakers per session is determined implicitly. On the other hand, *partitional* approaches to clustering operate differently. *Partitional* methods generate a clustering hypothesis by estimating a division of the data into $\widetilde{k}$ clusters by operating in a non-hierarchical, and non-nested fashion. These *partitional* methods are powerful clustering tools which, however, require the development of heuristics that allow for the *explicit* estimation of the number of speakers for their application to SD.

A *partitional* method explored in the recent years in its application to SD is that of SC [192]. SC formulation starts off by simple premise: same/different-class data points result in high/low similarities when compared to form an affinity-matrix. When operating upon perfectly separable data points, SC affirms that the eigendecomposition of a symmetric affinity-matrix leads to a set of orthogonal eigenvectors. At the same time, the number of non-zero eigenvalues indicates the dimensionality of the canonical decomposition of the complete data space. For the case of SD then, perfectly separable, speaker-discriminative representations would lead to an orthogonal representation of the speaker space in which the representative dimensionality would correspond to the number

of speakers. However, it is known that the speaker-discriminative representations used in SD are far from independent. In such a scenario, the estimation of the necessary number of classes that represent a discriminative speaker space becomes problematic. In estimating the number of speakers by means of SC, multiple approaches in the literature have leveraged the information that lies in the relationship between eigenvalues [119, 120, 194]. In particular, SC has proven to be a reliable solution to clustering when using *highly* training-dependent speaker embeddings leading to, at the time of its publication in [8], state-of-the-art performance in the SD of telephony data.

Motivated by the results of the BK-based baseline SD system in Section 6.3.3, in which the AHC algorithm seemed incapable of estimating the optimal number of speaker clusters, the remainder of the work in this chapter explores the use of SC in its first application to the training-independent BK-based SD. The extent to which BK-based speaker representations in the form of cumulative vectors (CVs) are sufficiently discriminative to result in a space of reduced dimensionality rightly related to the actual speaker space is studied. The relationships between eigenvectors and eigenvalues are assessed and applied to the BK-based SD pipeline by: (i) substituting the AHC algorithm, (ii) proposing the coupling of the number of speakers estimation based on SC with the AHC algorithm, and (iii) developing an eigenvalue-based solution to the detection of single-speaker sessions.

6.4.2 Spectral clustering and BK speaker modelling

Spectral clustering is a popular clustering technique in the literature. Out of the implementations that exist within the so-called spectral methods [195], we explored in this work the approach proposed by Ng et al. [192]. The general idea of this implementation is to perform clustering using the eigenvectors corresponding to the top eigenvalues estimated from an *affinity-matrix* derived from the similarities between data points being clustered. The process can be summarised in three stages (following the notation proposed in [195]):

Pre-processing

Normalization and smoothing operations are applied to the affinity-matrix to carefully increase its homogeneity. A number of further refinements also applied prior to the eigendecomposition, in an attempt to highlight leading to improved SD performance [8,119, 120]. They are based on the temporal locality of speech data. Contiguous speech segments uttered by the same speaker should have similar speaker-discriminative representations and hence similar values in the affinity-matrix.

In the case of BK-based speaker modelling the affinity-matrix is defined as follows. Given a test audio file, it is represented by a sequence of M CVs of the KBM dimension N, that compose a matrix J of dimension M-by-N. The M-by-M affinity matrix K is determined using the cosine similarity so that:

$$K = 1 - D_{cos}(J, J^T) \tag{6.1}$$

The pre-processing steps applied to the affinity matrix K are illustrated in Figure 6.2 which shows an example of the process computed for DIHARD development set file *DH_0028.wav* (with the original affinity matrix K presented in Fig. 6.2a). They include the following operations:

- **Gaussian blurring [196]**: Applied with standard deviation σ, the goal of this low-pass filter is to smooth the affinity matrix to reduce inconsistencies resulting from noisy representations, and filter out high-frequency components (Fig. 6.2b). The application of this technique may result counter-intuitive when thought of in an oracle manner: *perfectly* discriminative speaker representations could transition rapidly from one speaker to another within the course of a conversation, and it would not be desirable for a diarization system to diminish the changes in the affinity matrix generated by these transitions. Its use here, however, relates more to its common role in the task of edge detection [197]. Speaker-discriminative representations are admitedly not perfect and instead typically exhibit a noise component, which degrades the clarity of the edges derived from transitions between speakers.

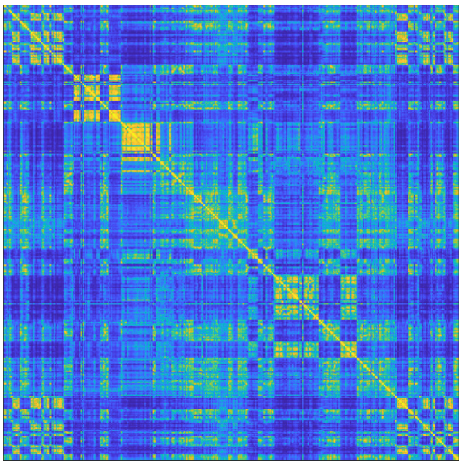
- **Row-wise thresholding**: Similarities below the p-percentile are discarded and set to 0 on a row basis to discard similarities that are very low. These are assumed to belong to comparisons between CVs that represent different speakers (Fig. 6.2c).

- **Symmetrization**: This operation restores the symmetry lost in the previous step, which is important for the process of eigenvector decomposition that will be applied to the final affinity matrix (Fig. 6.2d). When applied to the affinity matrix K, it results in a matrix where[2]:

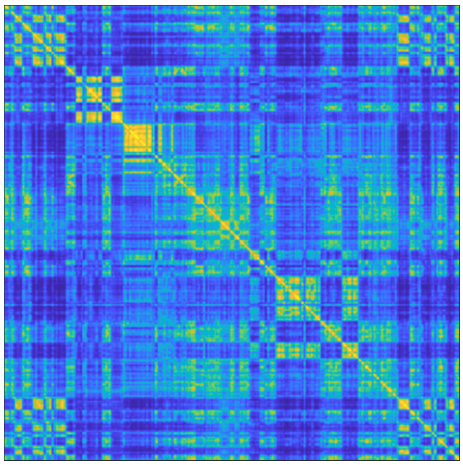
$$Y_{ij} = max(X_{ij}, X_{ji}) \tag{6.2}$$

- **Diffusion**: This normalization step was introduced in [8], which draw inspiration from the concept of Diffusion Maps [198](Fig. 6.2e). The intention behind its use is similar to that of the Gaussian blur in the first step, i.e. highlighting the

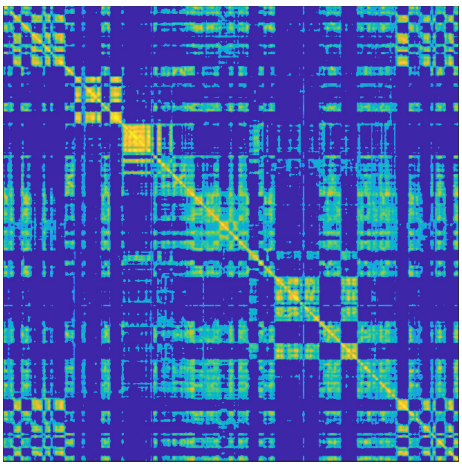
[2]Note we use a X and Y from hereon as input and output of the operation, respectively



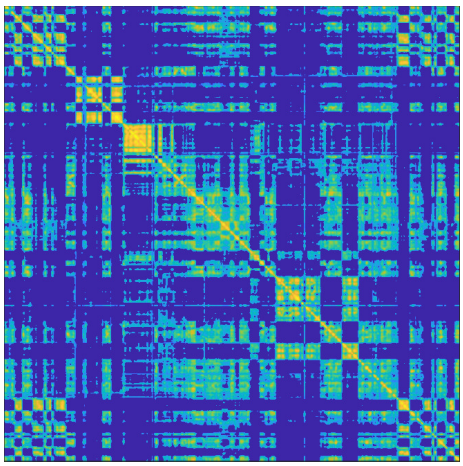
(a) Original affinity matrix



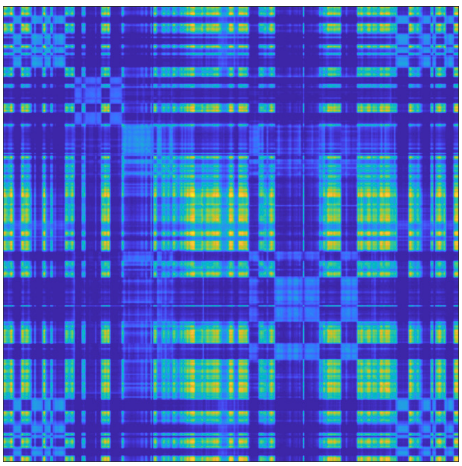
(b) Gaussian blur



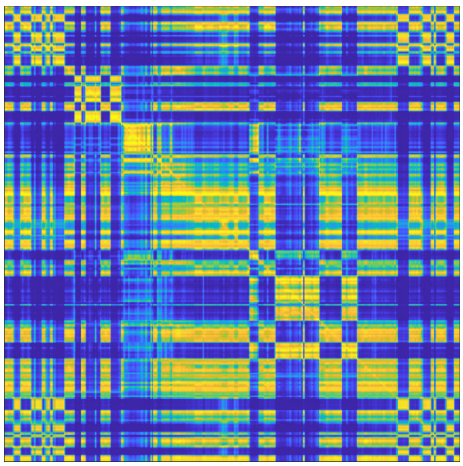
(c) Row-wise thresholding



(d) Symmetrization



(e) Diffusion



(f) Row-wise max normalization

Figure 6.2: Affinity-matrix of cosine similarities between CVs for the file *D_0028.wav*. Refinement operations are applied to the result of the comparison between the BK-based representations (a) to smooth and enhance patterns, presumably representing to speaker identities, visible in the similarity space.

differences in the speaker patterns of the now-enhanced affinity matrix. This is achieved through a simple multiplication between transposed affinity matrices as follows:

$$Y = XX^T \tag{6.3}$$

- **Row-wise max normalization**: A final re-scaling step that brings all activations in the affinity matrix to a more homogeneous level(Fig. 6.2f) before its eigendecomposition. The operation is applied for every row k of the affinity matrix, so that:

$$Y_{ij} = \frac{X_{ij}}{max_k(X_{ik})} \tag{6.4}$$

Spectral mapping

Eigenvalue decomposition is performed upon the enhanced affinity-matrix, returning eigenvectors and their respective eigenvalues. The motivation of this operation, introduced in Section 6.4.1, is that data observations, i.e. the pair-wise similarities between the CVs, live in a high-dimensional space of size M. Given perfectly separable observations, M may be approximated by means of a subspace of singular vectors of dimension $k \leq M$, where k is the oracle number of classes, e.g. speakers in an audio session. As the real number of speakers k is unknown in practice, techniques are necessary that estimate a number of clusters $\widetilde{k} \sim k$. A possible approach [199] relates to leveraging the ratios between consecutive, sorted in descending order eigenvalues $\lambda_1 > \lambda_2 > ... > \lambda_n$. This ratio may be referred to as the eigengap, and the value m which maximises it is used to estimate the number of clusters $\widetilde{k}$ so that:

$$\widetilde{k} = \arg \max_{1 \leq m \leq n} \frac{\lambda_m}{\lambda_{m+1}} \tag{6.5}$$

Grouping

Following the decomposition into eigenvectors and the estimation of the number of speakers $\widetilde{k}$, clustering can be done directly in the spectral domain. An M-by-$\widetilde{k}$ matrix of eigenvectors is used as a $\widetilde{k}$-dimensional representation of the M input CVs. In this work, and as means of alternative to the AHC algorithm, we explore the direct application of K-means clustering using the squared Euclidean distance. Besides, the results of the BK-based baseline system, presented in Section 6.3.2, showed a very large gap in performance between what the baseline AHC algorithm could potentially achieve (assessed through an experiment with an oracle estimation of the number of speakers) and what the automatic clustering selection mechanism could generate. In a second set of experiments presented

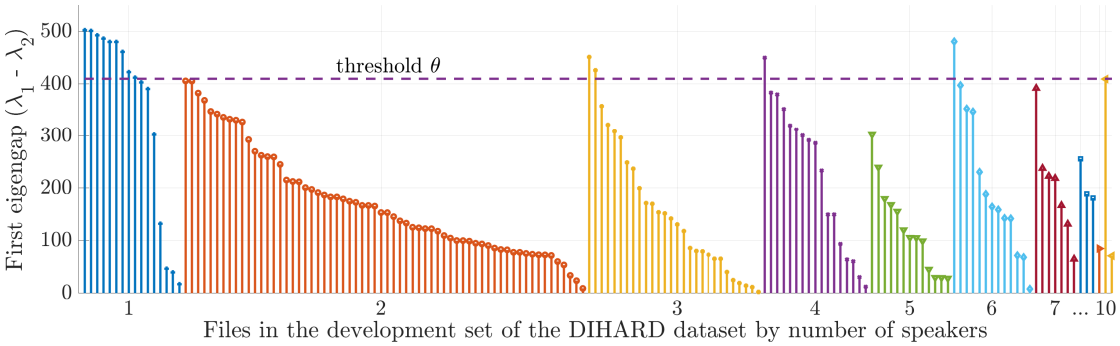


Figure 6.3: The eigengap resulting of the subtraction between the two first eigenvalues $\lambda_1 - \lambda_2$ is used as a measure to detect 1-speaker sessions. Sessions in the DIHARD development set are organized by number of speakers. A majority of the 1-speaker sessions in the development set are separable under our thresholding criterion.

in this work, the leveraging of SC is then also proposed as a stand-alone number of speakers estimator. The SC-based estimated number of speakers $\widetilde{k}$ may then be coupled with the fully-fledged AHC process, where it substitutes the baseline stopping criterion method.

6.4.3 Single-speaker detection

The problem of single-speaker SD is not widely explored in the literature. However, the DIHARD challenge raises this important issue by providing a *LIBRIVOX* audiobook domain containing only single speakers, in addition to a few other single-speaker files belonging to different domains. It is therefore important to accurately detect such audio streams, which otherwise can impact significantly on diarization performance. Mistakes in terms of number of speakers estimation are one of the most likely sources of error in SD systems. However, the impact of under-clustering single-speaker sessions are particularly meaningful, as they result in an excessive degradation to the performance with regard to the real impact on the overall performance of a SD system.

In consequence, we designed a mechanism for single-speaker detection that could continue leveraging the information derived from SC. To do so, we tackle the problem of single-speaker session detection using the gap between the first two biggest eigenvalues λ_1 and λ_2. The system performs single-speaker detection from pre-clustering according to the thresholding of the eigengap between the two largest eigenvalues. However, differently to the method to number of speaker estimation defined in Equation 6.5, the eigengap for single-speaker detection, illustrated in Figure 6.3, is considered as the subtraction between the eigenvalues, so that when $\lambda_1 - \lambda_2$ exceeds a threshold θ, then the number of clusters is forced to 1. Figure 6.3 presents the $\lambda_1 - \lambda_2$ score (Y-axis) for the DIHARD development

set files sorted by the number of speakers in a session (X-axis). Also illustrated is the empirically optimised single-speaker threshold θ. It is evident that, whilst not perfect, 1-speaker sessions are mostly separable by means of the threshold θ from the speech files containing more speakers. These detected files will result in 0% DER, and have a positive impact in the overall performance.

6.5 Experimental setup

This section elaborates on the dataset in Section 6.5.1, and feature extraction in Section 6.5.2. A description of the KBM and CV extraction configuration is given in Section 6.5.3, while clustering parameters are described in Section 6.5.4. Metric details are presented in Section 6.5.5.

6.5.1 Dataset

All experimental work reported in this chapter was performed with the standard DIHARD database [14,200,201]. The development set contains 164 audio documents from 9 different domains. The test set contains 172 audio documents from domains that comprise those of the development set and extra unseen scenarios, e.g. restaurant conversations. All results correspond to the use of ground-truth VAD annotations, i.e. track 1 of the DIHARD challenge.

6.5.2 Feature extraction

Baseline acoustic features are MFCCs comprising 19 static coefficients computed from windows of 25ms with 10ms overlap and with a filterbank of 20 channels. ICMC features use longer windows of 128ms, also with 10ms overlap.

6.5.3 KBM and cumulative vector parameters

The KBM is determined from a pool of Gaussians, each estimated using windows of between 0.5 to 2 seconds duration set dynamically so as to ensure a minimum of $G = 1024$ components. The size of the KBM N after Gaussian selection is set to an empirically optimised percentage of the number in the original pool. For the baseline system $\alpha = 85\%$ is considered following our work in [155]. As a result from the analysis presented in Section 6.3 (and illustrated in Fig. 6.1) α is finally set to $\alpha = 10\%$ for the remainder of the experiments.Segment CVs are estimated using 3s windows with 2s overlap. The top number of Gaussians per frame is set to $M = 5$.

6.5.4 Clustering parameters

AHC clustering is initialised with $N_{init} = 25$ clusters. Based on the distribution of the number of speakers per document on the development set, the maximum number of output clusters is set to 10.

For SC and the operations described in Section 6.4.2, some parameters are optimised using the development set. First, with regard to the refinement operations the standard deviation σ used in the Gaussian blur operation is set to $\sigma = 1$. The percentile $p = 40$ is determined to be the most effective for row-wise thresholding. Second, only eigenvalues larger than a threshold $\delta = 2.1$ are used to compute eigengaps that are used to decide the number of clusters. This is done to mitigate the effect of the so-called *noise* eigenvalues that are meaningless in the eigenspace obtained from the affinity matrix. If not limited, these values may produce anomalously large eigengaps, resulting in excessive clusters. Finally, the single-speaker detection threshold is set to $\theta = 410$. While this threshold, illustrated in Figure 6.3, includes some false alarms from files that have more than 1 speaker in the development set, this value offered the best trade-off between detection and performance.

6.5.5 Evaluation

System performance is was officially assessed using two different metrics. The primary metric was the standard diarization error rate (DER) with no forgiveness collar. Intervals containing overlapping speech regions were also scored. On the other hand, a secondary metric derived from frame-wise mutual information (MI) [14]. MI considers system evaluation from the stand-point of clustering evaluation. Both diarization reference and hypothesis thus compared as to score the mutual information in bits between the two labelings.

6.6 Results

Results presented in Table 6.1 show diarization performance measured in terms of the DER for both development and evaluation sets. Experiments using the baseline configuration (lines 1-2) as described in Section 6.3.2 use the same configuration as in prior work [155], with a relative KBM size of $\alpha = 85\%$, whilst comparing two different front-ends. Performance is very poor and non-conclusive, and only showing a good result on domain D2 of court recordings. However, the analysis elaborated in Section 6.3.2 motivates the choice of the front-end for the subsequent experiments. Figure 6.1 shows ICMC features lead the baseline AHC algorithm to a better performance when an oracle

stopping criterion is used. At the same time, ICMC features show optimum performance for a KBM size of $\alpha = 10\%$. Consequently, ICMC coefficents and $\alpha = 10$ are used in the remainder experiments (lines 3-5).

The results in the following sections explore the enhancements proposed in Section 6.3.3. Following the difference between the oracle and automatic estimation of the number of speakers presented in Section 6.3.2, it was decided to put the focus on the clustering stage of the BK-based SD system, as well as the approach to estimate the number of speakers. In particular, the results derived from exploring SC as a complete clustering approach are in Section 6.6.1. The results of using SC as a number of speakers estimator is described in Section 6.6.2. Section 6.6.3 describes results using the single-speaker detector approach. Finally, Section 6.6.4 presents some conclusions about the final system and its performance.

6.6.1 Spectral clustering upon CVs

Results in line 3 of Table 6.1 are obtained when using SC in place of AHC and elbow cluster selection. Prior to its use, refinement operations as described in Section 6.4.2 are applied to the affinity matrix derived from the M CVs of a session. This system significantly outperforms the BK-based baseline SD system by applying a K-means clustering upon the truncated eigenvectors of dimension $\widetilde{k}$. The number of partitions $\widetilde{k}$ is estimated as explained in Section 6.4.2. The use of SC upon CVs leads to a 29% relative reduction in DER over the baseline for the evaluation set. The first set of results including SC-derived information after the baseline analysis verifies the intuition that lies behind the application to our problem of SC: the process of eigendecomposition of the affinity matrix is capable of retrieving eigenvectors that are capable, upon truncation, of representing the speaker identities in the session. These results are proof of validity of the application of SC to a BK-based SD pipeline, constituting a domain-robust clustering alternative to AHC.

6.6.2 Spectral clustering as a number-of-speakers estimator

The performance of the baseline system reported in Section 6.3.2 highlighted not only the poor performance of the elbow criterion as number-of-speakers estimator for the DIHARD dataset, but also the much better performance generated by AHC if an optimum clustering solution could be retrieved. Given that the approach to SC applied here estimates the number of speakers $\widetilde{k}$ through an independent step in the clustering process, we could leverage this information and use the estimated number of speakers $\widetilde{k}$ as a stopping criterion for the AHC algorithm. The results obtained are reported in line 4 of Table 6.1.

	Development									Eval.
Systems	D1	D2	D3	D4	D5	D6	D7	D8	D9	ALL	ALL
1. MFCC / AHC / elbow (baseline)	59.64	**8.36**	44.35	46.38	28.34	46.97	46.99	66.69	56.75	44.47	48.31
2. ICMC / AHC / elbow	44.85	9.37	46.05	46.58	24.39	49.49	46.02	66.97	59.63	44.85	48.70
3. ICMC / SC	48.68	17.31	17.85	31.02	**11.36**	**23.65**	43.04	27.31	45.16	30.13	34.29
4. ICMC / AHC / $SC_{\#spk}$	**43.78**	14.19	**9.70**	**27.48**	12.71	23.99	**42.24**	11.22	38.33	25.77	30.44
5. ICMC / AHC / $SC_{\#spk}$ / 1-spk	**43.78**	14.19	11.02	**27.48**	12.71	23.99	43.55	**5.36**	**38.24**	**25.56**	**29.33**

Table 6.1: Speaker diarization performance in terms of diarization error rate (DER, %) of the baseline system and after incorporating the proposed enhancements, on the development and evaluation sets. DER is also broken-down by domain for the development set (D1: SEEDLINGS, D2: SCOTUS, D3: DCIEM, D4: ADOS, D5: YP, D6: SLX, D7: RT04S, D8: LIBRIVOX, D9: VAST).

Performance improves again, this time giving a relative improvement of 37% over the baseline system. This approach verifies the robustness of the AHC solution used in the baseline system, highlighting the fundamental importance of estimating reliable the number of speakers in a SD pipeline. The reason as to why this approach outperforms that of the previous experiment (line 3 of Tab. 6.1) may be related to a more robust initialisation of the AHC algorithm. The system explained in Section 6.6.1 performs k-means directly upon truncated eigenvectors derived from CVs modelled on segments of 3s of speech data. However, here, initial clusters in the form of CVs are extracted from speech contents that are much larger, thereby possibly leading to less errors in the clustering process.

6.6.3 Evaluation of the single-speaker detector

After improving the BK-based SD system performance by means of combining the baseline AHC algorithm with a number-of-speakers estimator based on SC, we now evaluate the effect of the single-speaker session detector proposed in Section 6.4.3. The result, presented in line 5 of Table 6.1, achieves a slight increase in the development set, but translates to a benefit of over 1% DER with regard to the previous best performance on the evaluation set. Its effectiveness allows us to achieve a relative improvement of almost 40% over the baseline performance.

6.6.4 Domain-based performance

Table 6.1 also shows granular results for each of the 9 domains D1-D9 (consult Section 6.2 for domain details) contained in the development set. The results presented in this section show improvements in diarization performance for most domains through application of system enhancements derived from SC in its first application to BK-based SD. The exception is D2 (SCOTUS, composed of recordings at court sessions), for which the baseline system performs best. This is attributed to the tendency of the SC selection algorithm to underestimate the number of clusters. However, the decrease in performance in this domain is small when compared to the gains achieved in other domains such as D3 or D5. At the same time, errors in some domains remain high, i.e. D1 and D7. D1 corresponds to the SEEDLINGS domain that comprises children at very early stages of their oral communication development. It is clear that SD technologies are not completely ready for this kind of task and that further work is required in order to tackle these domains more reliably. On the other hand, D7 consists of meeting domains from the NIST RT evaluations, widely acknowledged as difficult in the literature [191]. Of particular note are improvements for D8. Documents corresponding to this domain contain only a single-speaker where the novel single-speaker detection mechanism is

Evaluation set	DER	MI
Team A [99]	23.73	8.44
Team B [202]	24.56	8.47
Team C [117]	25.07	8.46
Team D [86]	26.02	8.35
Team E [203]	26.90	8.34
Team F	27.61	8.33
Team G	28.52	8.32
EURECOM [180]	**29.33**	**8.33**
Team I [204]	32.76	8.29
Team J [205]	33.15	8.39
Team K	33.79	8.14
Team L	36.73	8.18
Team M	37.46	8.04

Table 6.2: Comparison in performance of the evaluation set for the best submission of each team in the final classification. Results are reported in terms of the two official metrics of the challenge, i.e. diarization error rate (DER, primary metric) and mutual information (MI, secondary metric). The submission reported in this chapter obtained a mid-table result and compares favourably to the best performing systems considering the extensive amounts of training data and computing time required for the training phase of most other approaches.

especially effective in reducing the error rate.

6.6.5 Results in the official DIHARD classification

Results for the DIHARD challenge are presented in Table 6.2. These correspond to performance on the evaluation set, and have been filtered to show only the best submissions of each respective team. Performance is presented in form of the DER and MI which were used, respectively, as primary and secondary metrics. Results in the remainder of this section are, however, discussed exclusively with regard to the DER.

Direct comparisons are difficult across systems on account of differences not only in the SD pipeline used by each competitor, but also differences in the training data employed. However, the results reported by the top performing systems make for an interesting analysis. Team A [99] obtained a final DER of 23.7% achieved using a final Variational Bayesian (VB) resegmentation step without which the DER would be 25.5% DER. Team B [202] used speech denoising that provided a rough improvement of 0.7% DER on the development set. Assuming that a similar effect applied to the evaluation

set, its performance without denoising would degrade to a DER of 25.2%. Team C [117] achieved a DER of 25.07% DER after merging data and pseudo-labels derived from the evaluation set with the development data for training, without which they would have achieved a DER of 25.3 %. These results point to the existence of an apparent lower bound of performance in the evaluation set of around 25-26% DER when additional, speaker modelling independent, finely tuned processing steps are not included in the SD pipeline. In contrast, a 29.3% DER is achieved by the BK-based SD submission presented here which, to the best of the author's knowledge, was the only submission from all competing systems that does not use any external training data. The independence of training data does, however, not come free of cost as the performance for EURECOM's system is a 3-4% worse in terms of DER than that achieved by competitors.

6.7 Summary

This chapter reports the improvements applied to the BK-based SD system in the context of EURECOM's participation [180] to the first DIHARD challenge in domain-robust SD. While the baseline system is shown to perform poorly, the enhancements reported in this chapter lead to substantial improvements. These enhancements include features extracted using a perceptually motivated, variable spectro-temporal decomposition. While they were already discussed in Chapter 4 in the context of broadcast news, here their contribution to BK speaker modelling is verified in the context of the challenging multi-domain DIHARD dataset. Additional enhancements are a robust approach to cluster selection based upon spectral clustering and a mechanism designed to detect single-speaker segments. When combined, these enhancements bring a relative reduction in the diarization error rate of almost 40% over the baseline system. Performance, although lower than that of top-ranked systems, still compares favourably. This is especially so given that the proposed system requires no background data and is highly efficient, with execution times in order of 63 times faster than real time when running on a consumer-grade desktop computer.

With respect to the goal of domain-robustness, the proposed system based on BK modelling is a ready-to-run or off-the-shelf solution to SD. The estimates of SD performance reported in this chapter are likely to be reasonably reliable measures of performance if the same system were to be tested using data collected in other domains; the system is not dependent on optimisation using domain-specific background data beyond that in the development set and is instead tuned automatically at runtime. This is seen as a significant advantage over competing systems. This quality of the BK-based approach should be of appeal to practical applications of SD technology which is, after all, often an enabling technology rather than the final application.

Chapter 7

System combination

While stand-alone BK-based diarization has proven to be an efficient, domain-robust, competitive alternative to state-of-the-art systems, its performance stands a step behind some competing techniques. BKs are a fundamentally different approach especially with regards to solutions based on neural embeddings. Such different solutions may then be complementary. The Albayzin 2018 Speaker Diarization Challenge provided an ideal opportunity to investigate this research hypothesis. It was explored in the context of the ANR ODESSA joint submission.

This chapter presents and compares different state-of-the-art neural embeddings systems using the Albayzin 2018 Speaker Diarization Challenge dataset. The complementarity of a BK-based speaker diarization (SD) system is then assessed by means of different approaches to SD system combination in the context of *closed-set* and *open-set* training conditions. The remainder of this chapter is structured as follows. Section 7.2 details the processing blocks which compose the different diarization solutions. Section 7.3 introduces the problem of SD combination and the two particular techniques explored in the work reported here. Sections 7.4 and 7.5 report the experimental setup and systems with results. Finally, Section 7.6 provides a summary of the work and core findings.

7.1 Motivation and context

The previous chapters of this thesis explore enhancements that were applied to different processing modules that constitute the baseline SD system (defined in Section 3.7). These enhancements all bring substantial improvements to performance. In some cases, despite the advantages of the BK-based system, and depending on the scenario, it can still fall

slightly behind the very latest deep-learning solutions based on speaker embeddings (shown by results reported in Chapter 6).This tends to be the case when large amounts of matched, external training data are available.

However, the fundamentally different approach to SD embodied by BK speaker modelling raises the question of whether BK-based and neural network-based diarization systems can be complementary. The response is not obvious since, thanks to training based on extensive external data, neural network-based embeddings are expected to be more discriminative than BK-derived representations. Their combination with BK-based approaches might then not be beneficial. However, since BK-based solutions leverage the modelling of the in-session acoustic space, the combination may still help as an aid to clustering. The work presented in this chapter aims to determine whether or not the two technologies can be successfully combined.

The answer to this question falls within the problem of system combination/fusion. In contrast to other problems, the fusion of SD systems and/or their outputs is a especially challenging task due to the variability in both their fundamental operation and their outputs. This is probably one reason for why SD system fusion has not attracted significant interest in the research community. Notable exceptions include [206, 207] which provide a starting point for the work reported here.

The context of the research reported here relates to the ANR ODESSA submission to the Albayzin Speaker Diarization Challenge 2018, results of which were published in [208]. Albayzin evaluations cover a range of speech processing tasks that include search on speech, audio segmentation, speech-to-text transcription, and SD. While the 2016 edition allowed us to perform some work upon the feature extraction stage of our baseline system (reported in Chapter 4), the 2018 edition [13] includes newly collected and transcribed audio content from the RTVE2018 database [209], composed of TV shows covering a wide range of topics from the Spanish public TV network. Further details of the dataset can be found in [13, 209]. Two training conditions were proposed. First, a *closed-set* condition permits the use of only provided training data. Interestingly, the audio files in the provided set are composed by content of TV shows that, while belonging to the broadcast news/programs domain, differ significantly from that in the development and test set. The closed-set condition thus presents a significant challenge to domain-dependent neural embeddings. The *open-set* condition permits the use of *any* kind of data. Here, neural embeddings are expected to deliver better performance since they can benefit from external training data. This is in contrast to the BK-based approach.

Two different SD fusion approaches were tested according to the training condition

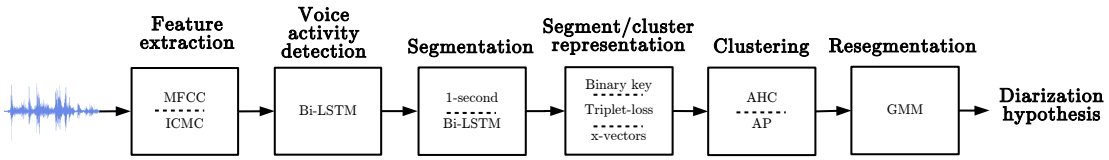


Figure 7.1: Diarization pipeline adopted by the proposed individual systems. Final systems were composed as combinations of the proposed modules.

and whether or not the systems under fusion provide synchronous outputs at different stages if the system pipelines. For the *open-set* condition, a common temporal resolution across systems is enforced and fusion is performed at the segment-level during clustering, allowing to jointly optimise the resulting system in an approach similar to that employed in [99]. For the *closed-set* condition, on the other hand, an arbitrary time resolution was used. This choice was made in order to mitigate the expectedly poorer performance of neural embeddings trained with limited data (and consequently respecting the systems independent and optimal configurations). Systems are combined at the diarization hypothesis level following a label merging approach inspired from [206].

7.2 Baseline system modules

This section introduces variations considered for each of the processing modules used to compose the diarization systems before their combination. The definition of the SD systems is presented as a composition of building blocks rather than complete pipelines as the fusion experiments included systems that vary by as little as a single module, e.g. two complete pipelines might differ only in a segmentation or clustering approach. These blocks, illustrated in Figure 7.1, include feature extraction, voice activity detection, segmentation, segment and cluster representation, clustering and resegmentation, and are defined in the following sections.

7.2.1 Feature extraction

Two different acoustic frontends were used (1st module of Fig. 7.1). They include (i) a standard Mel-frequency cepstral coefficient (**MFCC**) [19] front-end and (ii) an infinite impulse response - constant Q (IIR-CQT), Mel-frequency cepstral (**ICMC**) coefficient [160] front-end. The latter was used following the results reported in Chapter 4 [155] which proved beneficial to the BK-based SD system.

7.2.2 Voice activity detection and segmentation

All systems share a common voice activity detection (VAD) module (2nd block of Fig. 7.1). VAD is modelled as a supervised binary classification task (speech vs. non-speech), and addressed as a frame-wise sequence labelling task using a bi-directional long short-term memory (LSTM) network operating on MFCC features, following the work in [44]. As for segmentation (3rd module of Figure 7.1), two systems were explored: (i) a straightforward uniform segmentation which splits speech content into 1 second segments and (ii) segmentation via the detection of speaker change points. The speaker change detection (SCD) module is that proposed in [59]. Similarly to the VAD module, SCD is also modelled here as a supervised binary sequence labelling task (change vs. non-change) by means of a bi-directional LSTM similar to that of the VAD module. Both VAD and SCD systems were provided by LIMSI.

7.2.3 Segment/cluster representation

This section describes the speaker modelling techniques used for segment and cluster level representations (4th module of Fig. 7.1).

Binary key speaker modelling: BK speaker modelling is used as a segment/cluster representation in the form of cumulative vectors (CVs) as introduced in Section 3.4.

Triplet-loss neural embedding: This neural embedding architecture is the one introduced by LIMSI for speaker recognition in [57], further improved in [210]. The embedding space is generated by training based on the triplet loss paradigm [58] using a bi-directional LSTM recurrent neural network (RNN). In the generated Euclidean space, two sequences $\mathbf{x}_i$ and $\mathbf{x}_j$ of the same/different speaker(s) are expected to be close/far to/from each other according to their angular distance. These neural representations were provided by LIMSI.

x-vectors: This method [211] uses a deep neural network (DNN) which maps variable length utterances to fixed-dimensional embeddings. The network consists of three main blocks. The first is a set of layers which implements a time-delay neural network (TDNN) [212] which operates at the frame level. The second is a statistics pooling layer that collects statistics (mean and variance) at the utterance level. Finally a number of fully connected layers are followed by the output layer which has as many neurons as the number of speakers in the training dataset. The output layer neurons use soft-max activations. All other layers use ReLu activations. The network is trained to discriminate between speakers in the training set. Once trained, it is used to extract utterance-level embeddings for utterances from unseen speakers. The embedding is just the output of

one of the fully connected layers after the statistics pooling layer.

7.2.4 Clustering

Two different approaches to clustering (5th block of Fig. 7.1) are used and described here.

Agglomerative hierarchical clustering (AHC): The AHC algorithm is that described in the baseline system and detailed in Section 3.7.

Affinity propagation: As proposed for its application to speaker embeddings in [132], an affinity propagation (AP) algorithm [213] is the second clustering method. In contrast to other approaches, AP does not require a prior choice of the number of clusters. All speech segments are potential cluster centres (exemplars). Taking as input the pair-wise similarities between all pairs of speech segments, AP will select a set of exemplars and then associate all other speech segments to one of them. In our case, the similarity between the i^{th} and j^{th} speech segments is the negative angular distance between their embeddings.

7.2.5 Resegmentation

A GMM-based resegmentation process is performed to refine the time boundaries of the segments generated in the clustering step. It uses the approach defined in Section 3.7.

7.3 Fusion

The process of combining outputs derived from different systems is common practice in the closely related task of speaker recognition. Many submissions to the NIST SRE evaluations follow such approaches to combine a host of different classifiers [214, 215]. Speaker recognition trials involve a claimed identity, a test audio sample and a score. Fusion techniques are typically applied at the score level and can lead to substantial improvements in performance.

The problem of fusion of SD systems is considerably more complex. The difficulty steams from the temporal aspect inherent to SD. A SD pipeline, as illustrated in Figure 7.1, is commonly composed by several processing modules. Each one of these may operate differently but, more importantly in terms of fusion, they may produce unsynchronous outputs, e.g. different approaches to voice activity detection (VAD) could lead to different speech/non-speech boundaries. The same is true for other components, e.g. segmentation.

One solution to this problem is straight-forward: different modules and system can be configured to provide perfectly aligned labels. Outputs could then be fused at the score level following techniques based on the linear combination of systems similar to the approaches used for speaker recognition [216].

However, this strategy is not always convenient. While pre-computed VAD or segmentation inputs may be shared between different systems, enforcing such shared modules across individual system may lead to their suboptimal performance. One system could, for example, be optimised to operate upon a sequential segmentation, while another may employe a SCD-based mechanism. An alternative approach in this case is to perform fusion at the hypothesis level, e.g. upon speaker labels.

In regard of these scenarios, and depending on the level of integration existing between the individual SD systems, two approaches to fusion were explored. The first operates at the similarity-matrix level and is suited to the combination of SD tightly integrated systems producing aligned speaker labels. The second, inspired by the work in [206], operates at the hypothesis level, is applied to systems with arbitrary alignments.

7.3.1 Fusion at similarity-matrix level

Systems sharing the same VAD and segmentation boundaries can be combined at the similarity level as their segment-level representations are completely synchronised. In [99] fusion is performed by the weighted sum of the similarity matrices, i.e. the matrix resulting of the pair-wise comparison of all the segment/cluster level representations in a session, of two segment-aligned systems before linkage agglomerative clustering. In consequence, the fusion of the similarity matrices is considered only once per session. The approach to similarity-matrix level fusion proposed performs in a similar fashion, but different in that it performs fusion at every segment-to-cluster and cluster-to-cluster operation done in the AHC algorithm. Another difference lies in the method employed to estimate the number of speakers per session. The work in [99] operates upon an empirically optimised threshold to determine the stopping criterion of the AHC algorithm. In the approach proposed here, similarities matrices are also combined in the computation of the within-class sum of squares (WCSS) of each of the clustering solutions derived by the AHC algorithm. WCSS is then used in an unsupervised clustering selection method based on the elbow criterion (described in Section 3.6.2 and illustrated in Figure 3.6). In this way, the clustering process exploits information provided by each different system. These individual systems are weighted using a parameter α which is empirically optimised to minimise the DER for the development set. Given two M-cluster-to-N-segment similarity

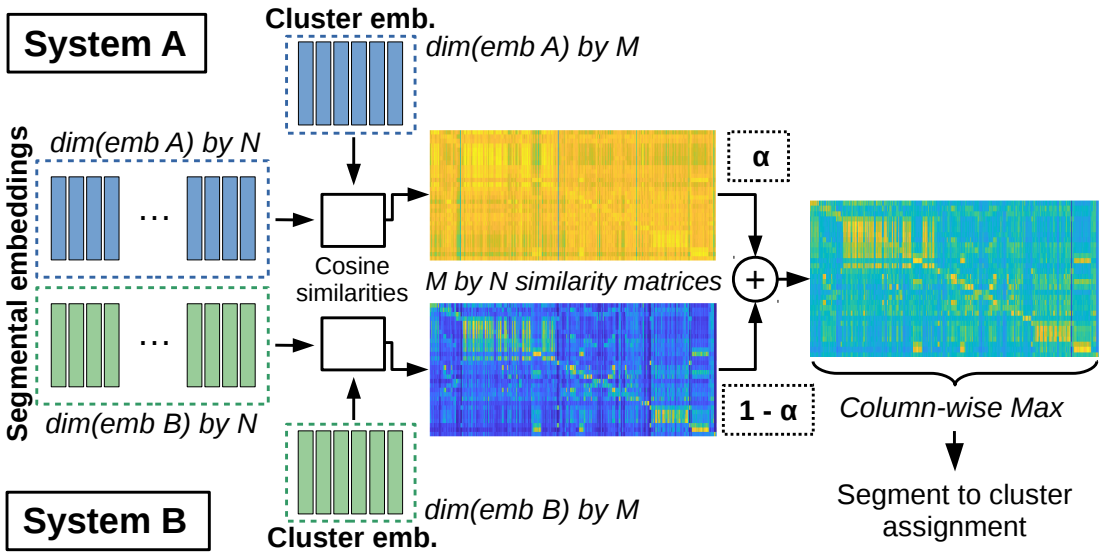


Figure 7.2: Illustration of the segment-to-cluster similarity-matrix level fusion. A tight integration in the processing of the system allows for segment-level representations to be perfectly aligned. Then they can be fused in the similarity-matrix domain to generate a more robust clustering and an improved number of speaker estimation.

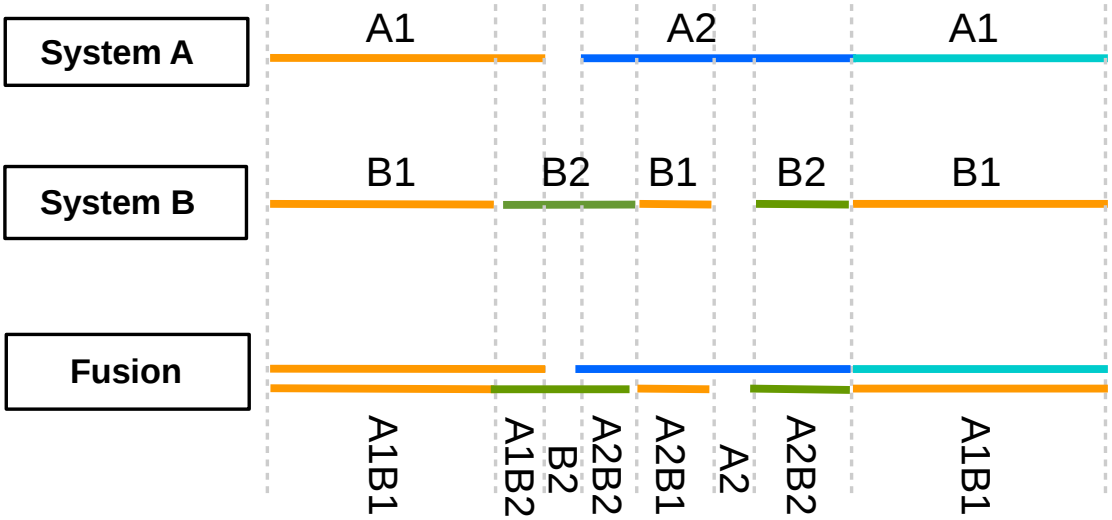


Figure 7.3: Illustration of the fusion of two diarization systems using the hypothesis level technique. When design constraints between systems make segment-level alignment inconvenient, this technique [206] allows to derive unique new labels by combining the labels of the individual systems. A last GMM-based resegmentation step generates a refined diarization output.

matrices M_1 and M_2, a fused similarity matrix M is calculated as:

$$M = \alpha M_1 + (1 - \alpha) M_2 \tag{7.1}$$

An example of this approach to fusion is illustrated in Figure 7.2.

7.3.2 Fusion at the hypothesis level

The combination of systems with totally independent (and non-aligned) SD pipelines is generally possible only at hypothesis level. In this work we explored hypothesis-level combination using the approach described in [206]. Given a set of diarization hypotheses, every frame-level decision can be merged to assign a new frame-level cluster label represented by means of the concatenation of all labels of the individual hypotheses. An example of this strategy is illustrated in Figure 7.3. This process results in a large set of hypothesised speaker clusters, generally greater than the number in any single system hypothesis. Clusters containing less than 15 seconds are discarded. The remaining speaker clusters are modelled then by means of Gaussian mixture models (GMMs) fitted to their respective acoustic features. A final feature level reassignment is then applied using the GMMs derived from the hypothesis-level fusion to obtain the final diarization hypothesis, following the approach to resegmentation defined in Section 3.7.

7.4 Experimental setup

This section gives details of the training data and the configuration of the different modules used in the combination of the different systems.

7.4.1 Training data

For the *closed-set* condition, training data consists of the *3/24 channel* database [12] of around 87 hours of TV broadcast programmes in the Catalan language.

For the *open-set* condition, two popular datasets were used:

SRE-data: It includes several datasets released over the years in the context of the NIST speaker recognition evaluations (SRE) [134], namely SRE 2004, 2005, 2006, 2008 and 2010, Switchboard, and Mixer 6. This dataset contains mostly telephone speech sampled at 8 kHz (whereby test data derived from models trained on SRE-data was previously down-sampled to 8 kHz).

VoxCeleb: The VoxCeleb1 dataset [217] consists of videos containing more than 100,000

utterances for 1,251 celebrities extracted from YouTube videos. The speakers represent a wide range of different ethnicities, accents, professions and ages, and a large range of acoustic environments. This dataset is sampled at 16 kHz.

7.4.2 Development data

Systems were tuned on approximately 15 hours of audio available at the 12 sessions that compose the partition *dev2* of the RTVE2018 database, provided with human-annotated transcriptions. These sessions belong to the Spanish TV shows "La noche en 24h" and "Millenium", each of roughly 1h duration and containing speech from an average of 14 speakers per session. For further details please refer to [209].

7.4.3 Modules configuration

Feature extraction: MFCCs are extracted with different numbers of coefficients depending on the subsequent segment representation: 23 static coefficients for x-vector, and 19 plus energy augmented with their first and second derivatives for triplet-loss neural embeddings. The BK-based system uses 19 static ICMC features [160]. Finally, the resegmentation stage uses 19 static MFCC features.

Segmentation: When a 1-second homogeneous segmentation is applied, segment level representations are extracted from 3-second long speech segments. This is done by means of including the preceding and following 1-second speech segments to the 1s segmentation window. Alternatively, when an explicit SCD-based segmentation is applied, segment level representations are extracted from the length derived by the SCD system.

Segment representation: For BK speaker modelling, the cumulative vector (CV) dimension is set to $\alpha = 40\%$ (after optimisation on the development set). Gaussians are learned on a sliding window of 2 seconds to conform a pool with a minimum size set of 1024. The x-vector system uses the configuration employed in the Kaldi recipe for the SRE 2016 task[1]. Data augmentation by means of additive and convolutive noise is performed for training. The dimension of the embeddings is 512, which was later reduced to 170 using *linear discriminant analysis* (LDA). While *probabilistic linear discriminant analysis* (PLDA) [218] is normally used as a scoring tool for x-vectors, the limited amounts of in-domain data for PLDA training led to inconsistent performance. Alternatively, the LDA-reduced representation of x-vectors are compared by means of the cosine distance. For triplet-loss embeddings, trained at LIMSI, and because of the lack of global identities in the Albayzin dataset, triplets are only sampled from intra-files

[1] https://github.com/kaldi-asr/kaldi/tree/master/egs/sre16/v2

for the *closed-set* condition. Thanks to the consistent speaker labels used in Voxceleb, triplets are also sampled from inter-files for the *open-set* condition.

Clustering: AHC is initialised to a number $N_{init} = 30$ clusters higher than the number of expected speakers based on the development data. The parameters of AP clustering such as the preference and damping factor were tuned by LIMSI on the development set with the chocolate toolkit[2].

Resegmentation: It is performed with GMMs with 128 diagonal-covariance matrices. Likelihoods are smoothed using sliding windows of 1s.

7.5 Results

Table 7.1 summarises the results obtained on the development and evaluation sets of the RTVE2018 database. Systems are formed from the combination of the modules presented above. Note that systems are also referred to here as *submissions* following the naming convention in the context of the Albayzin 2018 Speaker Diarization Challenge. All systems share the same VAD module, implying that the speech/non-speech segmentation is identical for each system. Its error rate was 1.9%, composed of a missed speech rate of 0.3% and false rate of 1.6% . Results are presented for *closed-* and *open-set* conditions. The DER is used for assessment and uses a 0.25s standard forgiveness collar. Results are presented separately for *closed-* and *open-set* conditions in Sections 7.5.1 and 7.5.2 respectively. Performance is also compared to that of other participants, and conclusions are drawn with regard to the influence of the combination methods in Section 7.5.3.

7.5.1 Closed-set condition

Individual systems

The first system $C1_c$ is that which employs CVs derived from BK speaker modelling. It uses ICMC features, 1-second uniform segmentation (following the results reported in Chapter 5), and AHC clustering. Alternatively, a stand-alone neural based approach is used in $C2_c$, comprising MFCC features, a bi-directional LSTM-based SCD, triplet-loss neural embedding representation (EMB) and AP clustering. The DERs on the development set were 12.3% and 14.1%, for systems $C1_c$ & $C2_c$, respectively. The performance is worse on the evaluation set, decreasing to DERs of 30.1% and 37.6%.

[2]https://chocolate.readthedocs.io/

Condition	Sys.	Features	Segmentation	Segment rep. / train data	Clustering	Fusion	Dev.	Eval.
Closed-set	P_c	-	-	-	-	$C1_c$,$C2_c$, **Hyp-level**	**10.17**	**26.67**
	$C1_c$	ICMC	1-second	BK / -	AHC	-	12.33	30.13
	$C2_c$	MFCC	Bi-LSTM	EMB / 3/24 data	AP	-	14.10	37.65
Open-set	P_o	-	1-second	-	AHC	$C1_c$,$C1_o$,$C3_o$, **Sim-level**	7.21	25.99
	P_p	-	1-second	-	AHC	$C1_c$,$C1_o$, **Sim-level**	**7.70**	**18.71**
	$C1_o$	MFCC	1-second	x-vector / SRE-data	AHC	-	9.29	20.28
	$C2_o$	MFCC	Bi-LSTM	EMB / Voxceleb	AP	-	11.46	36.75
	$C3_o$	MFCC	1-second	EMB / Voxceleb	AHC	-	26.68	50.61

Table 7.1: Summary of ODESSA Primary (P) and contrastive (C1/C2) submissions for the closed- and *open-set* (denoted by c and o subscript, respectively) conditions, including feature extraction, segmentation and training data used, segment representation, clustering and fusion. Performance (DER, %) is shown in the last column.

Hypothesis-level fusion

In the *closed-set* condition, and because of the segmentation mismatch of the best performing single systems, combination could not be performed at the similarity-matrix level. Hence, the combination was applied at the hypothesis level following the approach described in Section 7.3.2. The label combination procedure followed by a GMM-based resegmentation led to a lower DER than those of the two individual systems. The resulting system P_c is the fusion at the diarization hypothesis level of systems $C1_c$ and $C2_c$. The combined hypothesis decreases the DER to 10.17% for the development set and to 26.7% DER for the evaluation set. These results correspond to a relative improvement of 18% on the development set and 12% on the evaluation set with regard to the best individual system $C1_c$.

7.5.2 Open-set condition

Individual systems

In the *open-set* condition, systems based on neural embeddings are expected to perform better; they exploit large amounts of external training data. The SRE and Voxceleb databases were used for this purpose. System $C1_o$ used MFCC features, a 1-second uniform segmentation, x-vectors trained on SRE data and AHC clustering. Alternatively, system $C2_o$ is based upon triplet-loss neural embeddings. It resembles the *closed-set* $C2_c$ system, but where the training data was replaced with the Voxceleb data. The DERs for the development set are 9.29% ($C1_o$) and 11.46% ($C2_o$). For the evaluation set, performance decreases for $C1_o$ to 20.3% DER and $C2_o$ to 36.8% DER.

Similarity-matrix fusion

For the *open-set* condition the impact of combining neural embeddings-based systems with BK-based CVs was assessed through a similarity-matrix level fusion, possible thanks to their shared segmentation and VAD modules. The BK system is that defined above as $C1_c$ (we may use here the system as we do in the *closed-set* condition as BK modelling does not depend on any external training data). Two different system combinations were tested.

A first combination is that of system P_p (3rd system of *open-set* in Tab. 7.1) which included the BK-based diarization and the x-vector-based system $C1_o$. The fine tuning of the weighting factor α allowed for a tight integration in both the AHC algorithm and the number-of-speakers estimation method. The effect of α with regard to the DER is illustrated in Figure 7.4 (solid blue line). The contribution of the x-vector system $C1_o$ is

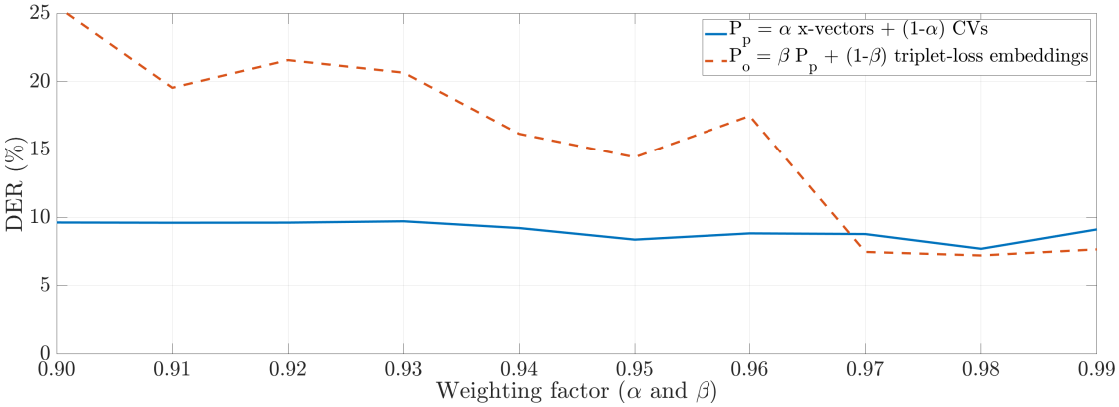


Figure 7.4: Diarization error rate (DER) for the development set of the RTVE 2018 Albayzin database as a function of the weighting factors α and β used for system combination at similarity-matrix level. α was used as a weighting factor to generate the system P_p which combines x-vectors and BK-based CVs. β was used to measure the inclusion of triplet-loss embeddings to the system P_p, leading to the system combination P_o.

weighted by α, whereas the BK-based system is weighted by 1-α. Following the notation used in Equation 7.1 this fusion configuration results in fused similarity matrices M_{P_p} calculated according to:

$$M_{P_p} = \alpha M_{C1_o} + (1 - \alpha) M_{C1_c}. \tag{7.2}$$

The optimal weight $\alpha = 0.98$ translates into a final system in which the x-vector system $C1_o$ contribution dominates. However, the inclusion of the BK-based CVs still increases performance. The DER is reduced from the 9.3 for system $C1_o$ on the development set to 7.7%, a relative improvement of 18%. For the evaluation set the system combination P_p delivers a DER of 18.7%, from the 20.3% obtained for the system $C1_o$ alone. While the improvement is smaller than that for the development set, it still constitutes a relative improvement of 8%.

The second approach to combination in the *open-set* aims to evaluate the benefit of a triplet-loss embeddings based system to the previously optimised fusion of x-vectors and CVs, fused system P_p. The motivation was not only related to the combination of neural embeddings and BK-based diarization, but also to the final joint-submission of the ODESSA partners in the *open-set* condition. In order to include the triplet-loss neural embeddings in the new system combination, its baseline segmentation based on a bi-directional LSTM and used on system $C2_o$ needed to be adapted to the 1-second homogeneous segmentation used by the other systems included in P_o. This was necessary in order to provide the alignment required for fusion at the similarity-matrix level. The

resulting system, labelled $C3_o$ in Table 7.1, suffered a degradation in performance with regard to $C2_o$. The DER decreased from 11.4% to a 26.68% for the development set and from 36.76% to 50.51% for the evaluation merely by the segmentation change applied to the system. Despite this decrease in performance it was decided to explore the extent to which a very lightly-weighted inclusion of a third system could be beneficial. Therefore the same approach to combination used for system P_p was applied here, this time as a function of a second weighting factor β. The resulting system is referred to as P_o. Following the notation in Equation 7.1, the similarity matrices M_{P_o} were derived according to:

$$M_{P_o} = \beta M_{P_p} + (1 - \beta) M_{C3_o}. \tag{7.3}$$

Optimisation results for β are illustrated in Figure 7.4 (dashed red line). Similarly as for system P_p, optimal performance is achieved for a weight of $\beta = 0.98$. Performance for the development set increased from 7.7% DER to a 7.2%, bringing a 7% relative improvement, which is acceptable considering the poorer performance delivered by $C3_o$ when using a suboptimal segmentation. The trend here, however, shows that the potential benefit of the inclusion of such a system was nonetheless more doubtful with regard to the standalone performance of P_p. Slightly smaller values of β would lead to large degradation in DER. Results for the evaluation confirmed the suboptimality of this fusion technique due to the inclusion of $C3_o$ leading to a DER of 25.99%.

7.5.3 Conclusions and results in the challenge

The official results of the Albayzin 2018 Speaker Diarization Challenge for the evaluation set are presented in Table 7.2 for 3 different scenarios: Table 7.2a presents the results on the *closed-set* condition for the submitted primary systems whereas Table 7.2b shows the equivalent for the *open-set* condition. Table 7.2c illustrates the results of the best system submitted by each participant in the challenge.

Closed-set training is a scenario in which a BK-based SD system benefits from its independence from external training data. In particular, the language mismatch between training (in the Catalan language) and testing (in the Spanish language) data subsets reinforces the relevance of the training-independent BK-based approach, even as to raise the question of which system is complementing which. In fact, the BK-based standalone system $C1_c$ outperformed the triplet-loss embeddings system $C2_c$ used for combination. As with regard to system combination, hypothesis-level fusion [206] allowed for separate systems may be tuned independently to their optimum performance

Team	DER
CP1 [219]	17.27
Ours (P_c)	**26.67**
CP3 [220]	26.70
CP4 [221]	32.20
CP5 [222]	34.66
CP6 [223]	39.09

(a)

Team	DER
Ours (P_o)	**25.99**
OP2 [223]	28.65
OP3 [224]	32.20
OP4 [225]	34.66

(b)

Team	DER
CP1 [219]	17.27
Ours ($C1_o$)	**20.28**
CX3 [220]	25.46
OX4 [223]	28.18
CX5 [222]	28.74
OX6 [224]	30.80
OX7 [225]	30.96
Ours (P_p)	**18.71**

(c)

Table 7.2: Official results for the evaluation set of the Albayzin 2018 Speaker Diarization Challenge in terms of DER (%). (a) and (b) report the results of the primary submissions on the closed- and *open-set* conditions, respectively. (c) presents the results obtained by the best systems per participant (primary or contrastive). Team name notation for other participants denotes its training condition (C is for *closed-set* and O is for *open-set*) and system (P for primary and X for contrastive).

before their combination. Results for the *closed-set* condition (Tab. 7.2a) highlight the complementary character of the approaches by obtaining a 2nd best place for the fused system P_c.

Results for the *open-set* condition shown in Tab. 7.2b show the benefit of using a similarity-matrix level approach to fusion. Our submission P_o including state-of-the-art x-vectors, CVs and triplet-loss embeddings ($C1_o$, $C1_c$ and $C3_o$ in Tab. 7.1), obtained the best result with a DER of 25.99% and the 1st place on the *open-set* condition.

Despite this result, performance of P_o was impacted by the poor performance of system $C3_o$ for the evaluation set. The potential of the fusion of a BK-based and x-vectors SD systems is nonetheless illustrated in the last row of Table 7.2c for system P_p. The inclusion of BK-based representations helps increase the robustness of a state-of-the-art x-vector system as is $C1_o$, which has a stand-alone performance of 20.28% DER. The resulting system P_p leads to a final DER of 18.71%, and an 8% relative improvement, achieving the 2nd best performance for all submissions to the challenge (Tab. 7.2c).

This improvement in performance provided by the BK-based SD system in this scenario demonstrates its capacity to complement state-of-the-art speaker embeddings in both hypothesis-level and similarity-matrix-level approaches to SD fusion. The work presented in this chapter suggests that different SD solutions might be capturing complementary information and points towards the benefit of SD system fusion. Since the BK-based

system is also training-data independent and may be deployed without any significant pre-optimisation, training or further adaptation, it is also a natural choice for system fusion whatever the scenario or data domain.

7.6 Summary

The work presented in this chapter shows that BK-based approaches to SD can be complementary to other techniques, specifically training-dependent speaker embeddings based on deep learning (DL). This was demonstrated by experiments that considered different approaches to SD system combination, work performed through the collaborative ANR ODESSA project and its submission to the Albayzin 2018 Speaker Diarization Challenge. Fusion strategies include a similarity-matrix level approach to fusion which supports the combination of segment-time-aligned SD systems. This method enables a tight integration between the different speaker representations employed at the cost of forced synchrony between SD pipelines. Alternatively, and drawing from an existing approach to SD system fusion [206], asynchronous systems, which allows for greater flexibility in their respective optimisation, can be fused using a label-merging technique which operates at the hypothesis level.

The fusion of BK- and DL-based SD systems leads to better SD performance. Submissions to the *closed-set* training condition of the Albayzins 2018 Speaker Diarization Challenge based upon a hypothesis-level fusion obtained 2nd position. On the other hand, when DL-based methods leverage large amounts of training data in the context of the *open-set* training condition, the combination with BK-based approaches by means of similarity-level fusion obtained 1st place. These results add further weight to the benefit of BK-based approach to SD which still remains training- and domain-independent; it provides an efficient means to enhance the performance of other techniques. When training-data is scarce, or not sufficiently well matched, however, BK-based solutions may still outperform more sophisticated techniques, e.g. those based on DL.

Part II

Low-latency speaker spotting

Part II of this concerns the use of speaker diarization within a real application. Motivated by the security and surveillance-related objectives of the ODESSA project, the work considers a new problem referred to as low-latency speaker spotting (LLSS). It wishes the detection, as soon as possible, of blacklisted individuals in multi-speaker audio streams. Chapter 8 presents the first formal definition of this task, and other contributions which allow for the integration of online SD with speaker detection. Additional contributions include protocols which support LLSS research using a publicly available database, and an initial solution using different ASV techniques. Chapter 9 proposes a modified LLSS system pipeline that frames diarization and speaker detection solutions at their heart. It leverages speaker models to guide the online diarization process by a novel selective cluster enrichment (SCE) process.

Chapter 8

Speaker diarization: integration within a real application

This chapter introduces a new task termed low-latency speaker spotting (LLSS). Related to security and intelligence applications, the task involves the detection, as soon as possible, of known speakers within multi-speaker audio streams. The chapter describes differences to the established fields of speaker diarization (SD) and automatic speaker verification and proposes a new protocol and metrics to support exploration of LLSS. These can be used together with an existing, publicly available database to assess the performance of LLSS solutions also proposed in the chapter. They combine online diarization and speaker detection systems. Diarization systems include a naive, over-segmentation approach and fully-fledged online diarization using segmental i-vectors. Speaker detection is performed using Gaussian mixture models, i-vectors or neural speaker embeddings. Metrics reflect different approaches to characterise latency in addition to detection performance. The relative performance of each solution is dependent on latency. When higher latency is admissible, i-vector solutions perform well; embeddings excel when latency must be kept to a minimum. With a need to improve the reliability of online diarization and detection, the proposed LLSS framework provides a vehicle to fuel future research in both areas. This collaborative work, published in [226], was undertaken within the context of the ODESSA project. The contributions of other ODESSA partners is gratefully acknowledged.

The remainder of the chapter is organised as follows: an introduction and a discussion of the motivations are given in Section 8.1. Related work is described in Section 8.2. A formulation of the LLSS task and metrics are given in Section 8.3. A first approach to LLSS is described in Section 8.4. Section 8.5 introduces the proposed protocol for LLSS

assessment. Experimental results are presented in Section 8.6. A final summary of the work is given in Section 8.7.

8.1 Introduction

An automatic speaker verification (ASV) system is usually tasked with determining whether or not an audio sequence contains a given speaker [4, 227]. Almost all work in the area, e.g. [228, 229, 230, 231], involves offline processing. This chapter reports work to develop a somewhat different system. In our task the ASV system is required to determine whether or not an audio sequence contains a given speaker *as quickly as possible*. We refer to this task as low-latency speaker spotting (LLSS).

The motivation relates to the needs of the security and intelligence services. These involve the rapid and efficient detection of known, target speakers from high volume audio streams. In such cases, rapid detection is needed in order to facilitate rapid reaction or response to potentially hostile intent; the first step subsequent to detection involves an agent listening immediately to the audio stream. While it is not the focus of our work, the LLSS task also relates to civilian and consumer applications involving voice-based personal assistants and speaker-dependent, but text-independent wake-up systems.

For the security/intelligence application, the cost of missing target speakers is high and the available resources to support human listening are limited. In this sense the appropriate metric for the assessment of solutions is similar to that used in the majority of related research [232, 233], namely the cost of detection (C_{det}) with the usual parameters. Here though, the emphasis on low-latency necessitates a two-dimensional metric which combines the cost of detection with the detection lag or latency.

The minimisation of latency has implications on the manner in which an audio sequence is processed. The LLSS task implies processing at a segmental level. While shorter segments will allow for detection with shorter latency, the associated reduction in data will naturally degrade reliability [234], inferring the need to strike a balance between latency and reliability. Furthermore, in our application there is also potential for multiple, competing speakers. Here too, then, there are differences between the existing research and the LLSS task. Solutions will likely combine ASV technology with some form of online segmentation or SD.

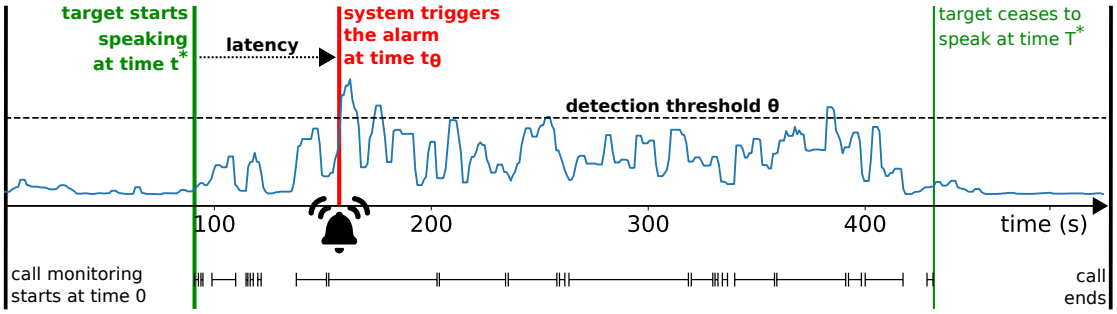


Figure 8.1: Low-latency speaker spotting (LLSS) systems aim to detect target speakers with the lowest possible latency.

8.2 Related work

The topics of SD and automatic speaker verification are closely related to the LLSS task. Speaker diarization [2] involves the clustering of speech recordings into speaker-homogeneous segments. In contrast to the LLSS task, SD is typically performed offline and with no prior information (e.g. number of speakers or speaker models). A number of online diarization [235, 236, 237, 238, 239] and speaker tracking [240] solutions have been reported. These use online speaker clustering algorithms [241, 242]. Only speaker tracking systems assume prior knowledge of target speakers but they do not consider latency.

Ideally, speaker recognition should be possible by using small amounts of speech. Unfortunately, with current technology, this is only possible if the text employed for enrolment and testing phases is constrained. This task is known in the literature as text-dependent speaker recognition [243, 244], and is often associated with specific applications, e.g. user-friendly human-robot interaction [216]. On-going research focused on keyword-spotting offers solutions that do not require more than a mere few seconds of speaker content [245], resulting in extremely low latencies.

However, text-related constraints are not suitable for certain scenarios, like surveillance, which motivate the majority of text-independent ASV research [227]. The text-independent ASV task tends to involve either single-speaker or two-speaker recordings. Research within the scope of the Speakers in the Wild (SITW) [246] initiative considers multi-speaker scenarios which necessitate some form of diarization as a precursor to ASV. Published research addresses only offline processing and the lack of speaker segmentation references means that the SITW database is ill-suited to the exploration of LLSS.

None of the prior work addresses all aspects of the LLSS task. Existing databases do not support the joint evaluation and optimisation of SD and text-independent recognition,

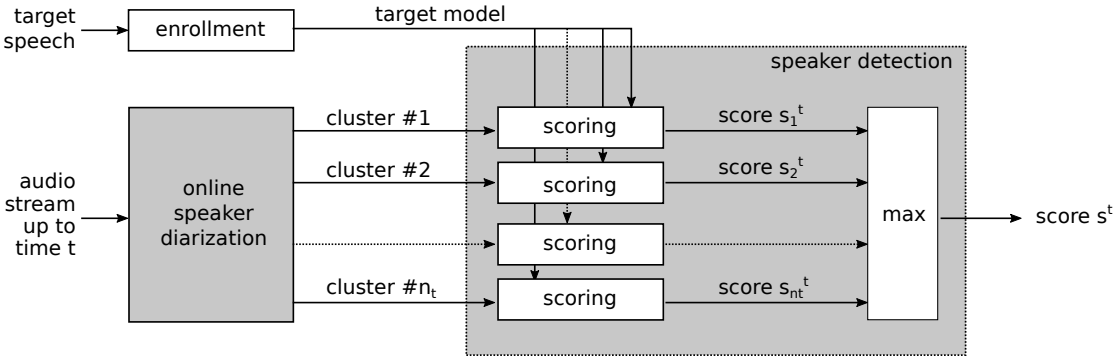


Figure 8.2: Common architecture to proposed LLSS solutions

nor the development of online, low-latency solutions. In addition, while existing databases can be adapted, there are no common protocols to support LLSS research.

8.3 Low-latency speaker spotting

This section provides a formal definition of the low-latency speaker spotting (LLSS) task[1] and outlines two different approaches to evaluate the latency achieved by potential solutions.

8.3.1 Task definition

The low latency speaker spotting (LLSS) task aims at determining whether or not an audio sequence contains a given speaker with the shortest possible delay. Figure 8.1 illustrates the sequence of an audio stream (e.g. an intercepted telephone conversation) during which a known, target speaker (for which example speech data is available) is active during the indicated segments. The target is active from time t^* but is detected only at time t_θ. The goal of the LLSS task is to detect the activity of the target speaker as soon as possible, i.e. to minimise the detection latency $t_\theta - t^*$.

Note that this is different from explicitly providing the speaker starting time t^*. If this value is needed in the context of a specific application, further automatic or manual processing may occur in order to refine t^* estimates once the target speaker has been detected. For a monitoring system, the audio stream may have been buffered and a security agent may listen to the stream after rewinding the audio by a few seconds, according to the typical detection latency; for a real-time human-robot interaction application, spotting the various users as quickly as possible is the main goal, regardless of the precise value of t^*.

[1]The particular contribution to this work of Hervé Bredin & Claude Barras is gratefully acknowledged.

An LLSS system may typically rely on regular log-likelihood ratio estimates (example blue profile in Figure 8.1) according to:

$$\Lambda(t) = \ln f(a_0^t|H_1) - \ln f(a_0^t|H_0) \tag{8.1}$$

where a_0^t is the audio from time $t = 0$ to time t and $f()$ is a conditional probability density given hypothesis H_1 or H_0, namely that the target speaker is either active in the audio stream at some point up to time t or not.

Given a detection threshold θ, the LLSS decision Γ at time t would then be:

$$\Gamma(t) = \mathbb{1}\left(\max_{\tau\in[0,t]} \Lambda(\tau) - \theta\right) \tag{8.2}$$

where $\mathbb{1}$ is the Heaviside function (returning 0 and 1 for negative and positive values, respectively). Note that the decision is irreversible once the threshold has been reached, even if Λ may later decrease. An ideal log-likelihood ratio estimator should thus return $\Lambda(t) < \theta$ for $t < t^*$ and $\Lambda(t) \geq \theta$ for $t \geq t^*$. In practice, $\Lambda(t)$ need not be produced periodically, but can be produced at arbitrary instances, leading to piecewise constant functions $\Lambda : \mathbb{R}^+ \mapsto \mathbb{R}$.

8.3.2 Absolute vs. speaker latency

An ideal LLSS system would trigger an alarm as soon as the target speaker starts speaking. In practice, this is not feasible as a certain amount of speech from the target speaker is needed before they can be recognised or 'spotted'. For instance, in Figure 8.1, the alarm is triggered at $t_\theta \approx 150s$ while the target speaker starts speaking at $t^* \approx 100s$, leading to an *absolute latency* δ of approximately $50s$.

In practice, the absolute latency δ will be influenced by the detection threshold θ. Low values of θ may lead to the alarm being triggered too early, before the target speaker starts speaking. For the sakes of evaluation (specifically the need to maintain a constant number of trials and to assign a latency to each), those trials are not marked as false alarms. Instead, their latency is bound to 0[2]. High values of θ may lead to the alarm not being triggered at all. In between, latency will likely increase monotonically with θ.

[2]Note that low values of θ would also lead to a high number of false alarms, making the system useless in practice. Such operating points lack practical interest.

More precisely, the *absolute latency* is defined as:

$$\delta_\theta = \max(t_\theta - t^*, 0) \tag{8.3}$$

where t^* is again the time at which the target starts speaking for the first time and t_θ is the time at which the alarm is first triggered.

In the case that the alarm is never triggered, t_θ is set to the time T^* in the audio stream at which the target speaker ceases to be active, giving:

$$t_\theta = \begin{cases} \min\left\{t \in \mathbb{R}^+ | \Lambda(t) > \theta\right\} & \text{if } \exists t \in \mathbb{R}^+,\ \Lambda(t) > \theta \\ T^* & \text{otherwise} \end{cases} \tag{8.4}$$

However, this definition may lead to arbitrarily high latency in the case, for example, that the first (possibly short) utterance of the target speaker is missed and the second utterance occurs long after. A more meaningful, alternative metric is the *speaker latency*, defined as the actual duration of speech uttered by the target speaker in the $[t^*, t_\theta]$ time range.

8.3.3 Detection under variable or fixed latency

For a given detection threshold θ, the value of either the absolute or the speaker latency δ_θ as defined in Eq. (8.3) will depend on the actual trial. If one does not constrain the maximal latency and lets the system use whichever latency gives the best detection performance (i.e. equivalent to $\Gamma(t)$ with $t \to \infty$), then this is referred to as a *variable latency* scenario. Detection performance and detection latency are then two complementary (but possibly contradictory) metrics. The average detection latency increases monotonically with θ, while the detection cost reaches its minimum value for a specific value of θ. Therefore, one may rely on curves displaying the detection cost as a function of δ to compare the performance of different systems.

However, averaging the latency across trials may in fact hide very different behaviours. Depending on the final application, we might prefer to evaluate the detection performance of a LLSS system at a given application-driven latency δ. In this *fixed latency* scenario, the system is expected to trigger an alarm during the $[0, t^* + \delta]$ time range. The detection performance of such a system may then be calculated using corresponding scores according to:

$$\lambda_\delta = \max_{t \in [0, t^* + \delta]} \Lambda(t) \tag{8.5}$$

Depending on the value of the detection threshold θ, the system will trigger an alarm if $\lambda_\delta \geq \theta$ whereas no alarm will be triggered if $\lambda_\delta < \theta$. Standard speaker recognition metrics then apply. They include the false alarm rate $FAR_\delta(\theta)$, the missed detection rate $MDR_\delta(\theta)$, the equal error rate EER_δ, and the detection cost $C^\delta_{det}(\theta)$ given by:

$$\begin{aligned} C^\delta_{det}(\theta) &= C_{\text{miss}} \times P_{\text{target}} \times MDR_\delta(\theta) + \\ & \quad C_{\text{false alarm}} \times (1 - P_{\text{target}}) \times FAR_\delta(\theta) \end{aligned} \tag{8.6}$$

8.4 LLSS solutions

This section describes a number of different solutions to the LLSS task. They share a common architecture depicted in Figure 8.2 which combines online SD with different approaches to speaker detection. At any time t, online SD provides a set of n_t speaker clusters $\{c^t_i\}_{1 \leq i \leq n_t}$. Speaker detection is then applied to compare the speech segments in each cluster c^t_i against a set of pre-trained target speaker models, thereby giving scores (or likelihood-ratios) s^t_i. A final score at time t is defined as the maximum score over all clusters: $s^t = \max_{1 \leq i \leq n_t} s^t_i$. The remainder of this section describes the two different online SD systems and three speaker detection systems explored in this work.

8.4.1 Online speaker diarization

Two different approaches to online SD are compared. Both rely on an LSTM-based voice activity detector (VAD) [59].

Segmental diarization: the first online diarization module does not perform any clustering: it relies simply on a segmental approach of a 3s sliding window with a 1s shift, and creates a new cluster at each step. Note that only speech content, often shorter than the complete 3s, is considered. This approach is denoted as segmental diarization in the rest of the chapter.

Automatic diarization: the second automatic system is based on i-vectors [231] and online sequential clustering using the same sliding window, a cosine similarity measure and an empirically optimized threshold to assign segments to existing clusters, or to create new ones. Should the score of a new segment produced from its comparison against the set of existing clusters fall below the threshold, then it will be assigned to a new cluster. Otherwise, it will be assigned to the cluster among the existing set corresponding to the highest score. Speaker clusters are represented by i-vectors extracted from the averaged sufficient statistics of their respective segments. The system uses 19 MFCC coefficients as a front-end, a universal background model (UBM) of 256 components and

a T matrix of rank 100, both learned from training data. i-vectors are length-normalised and whitened. All parameters were empirically optimised on the development set with according to the standard diarization error rate (DER) metric.

Oracle diarization: the performance of both online diarization systems is compared to that of an oracle diarization system in order to observe the impact of diarization errors on LLSS performance. The oracle system simulates the behaviour of an error-less, but still *online* system; it uses data from time zero to time t.

8.4.2 Speaker detection

The performance of three different approaches to speaker detection were explored. The systems considered are described in the following.

GMM-UBM: the first system is a standard, 256-component Gaussian mixture model with universal background model (GMM-UBM) [228], with a conventional MFCC frontend (the same as that used for diarization), *maximum a posteriori* model adaptation and log-likelihood ratio scoring.

i-vector: the second is an i-vector system [231] with a T matrix of dimension 100 and PLDA scoring [81] between target and test i-vectors, that uses a 100-dimensional speaker space and was trained on the same data as the UBM and the total-variability matrix. The front-end features are the same as that of the GMM-UBM system and the diarization system.

Neural embedding: the final system is based on the neural speaker embedding approach introduced in [57] and further improved in [210]. These were developed at LIMSI. Briefly, an LSTM-based neural network is trained to project speech sequences into a 192-dimensional space, using the triplet loss paradigm. Implementation details are identical to the ones used in [247]. The target (resp. cluster) model is the sum of all embeddings extracted from a 3s sliding window with a 1s shift over the enrollment data (resp. cluster). Resulting vectors are compared using the cosine distance.

8.5 LLSS assessment

None of the existing databases employed in either SD or speaker detection/verification are suited to the exploitation of the LLSS task. This section describes the steps taken to adapt an existing database for this purpose.

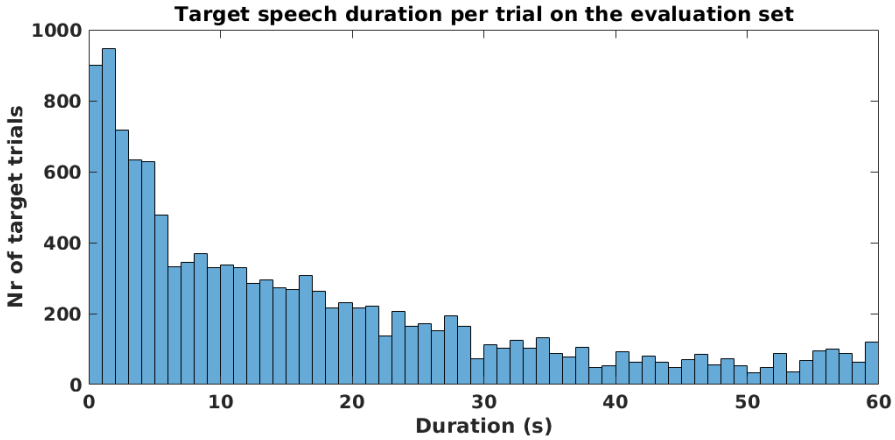


Figure 8.3: Distribution of target speech duration per trial for the designed test subset.

Set	# speakers	# models	# target	# non-target
Train	127	-	-	-
Dev.	22	121	9430	64451
Eval.	24	164	12250	102560

Table 8.1: LLSS protocol details: number of speakers, number of enrolled models, and number of target and non-target trials.

8.5.1 Database

The evaluation of LLSS solutions requires a large database of multi-speaker audio recordings and ground-truth speaker and segment level annotations. While several multi-speaker databases exist (e.g. the SITW database [246]), the Augmented Multi-party Interaction (AMI) meeting corpus [248] is widely used, publicly available and is provided with the necessary speaker and segment annotations. Consequently, it was adopted for all experimental work reported in this chapter.

The AMI database contains a set of audio meetings containing sessions of approximately 40 minutes and recorded across 3 different sites under different conditions and scenarios. As a consequence, speakers groups are disjoint in terms of site, while meetings collected at each site contain independent speaker groups with around 4 speakers each. There are approximately 4 meeting recordings for each group.

8.5.2 Protocols

Despite the use of a standard database, it was necessary to design new protocols to support the development and evaluation of LLSS solutions. Nonetheless, the standard

full-corpus[3] training, development and evaluation partitions are still respected. All experiments were performed using data corresponding to the mix-headset condition of the AMI meeting corpus.

Training data is used exclusively for background modelling. Speaker disjoint development and evaluation sets are both partitioned into enrolment and test subsets. Enrolment data is used to train target speaker models.

The single session which contains the greatest amount of speech from a given target speaker is used for enrolment. The speech from the target speaker is divided into N 60-second, overlap-free speech segment splits. A subset of these N splits is randomly selected as the data for the M different models generated for the target speaker. Since N varies across target speakers (due to varying quantities of data per speaker), M is set to the median of every N for each target speaker.

Testing content is generated from all the non-enrolment content for each given speaker and through sub-session splits of 1-minute duration. Each split contains speech from 1 to 4 speakers. While not ideal, under strict data constraints, the splitting of audio files serves to increase the number of trials and variability.

A single LLSS trial is similar in nature to a classical ASV trial; it involves an enrolled target model, a test sub-session, and a trial class (target/non-target). Target trials for a given speaker are defined by using all the test sub-sessions in which the target speaker is active. This leads to a distribution of target trials illustrated in Figure 8.3. The target speaker content per trial exceeds only rarely 30 seconds duration. Remaining sub-sessions correspond to non-target trials. The protocol described above results in the number of speakers, models, target and non-target trials illustrated in Table 8.1.

8.6 Experimental results

The performance of the proposed LLSS solutions is analysed in two different manners. The first analysis is in terms of fixed and variable speaker latency using detection metrics described in Section 8.3.3. Second, we analyse diarization influences upon LLSS performance.

8.6.1 LLSS performance: fixed latency

Plots in Figure 8.4 depict the evolution in EER for the evaluation set as a function of *fixed speaker latency* (latter part of Section 8.3.3). Separate plots are shown for

[3] http://groups.inf.ed.ac.uk/ami/corpus

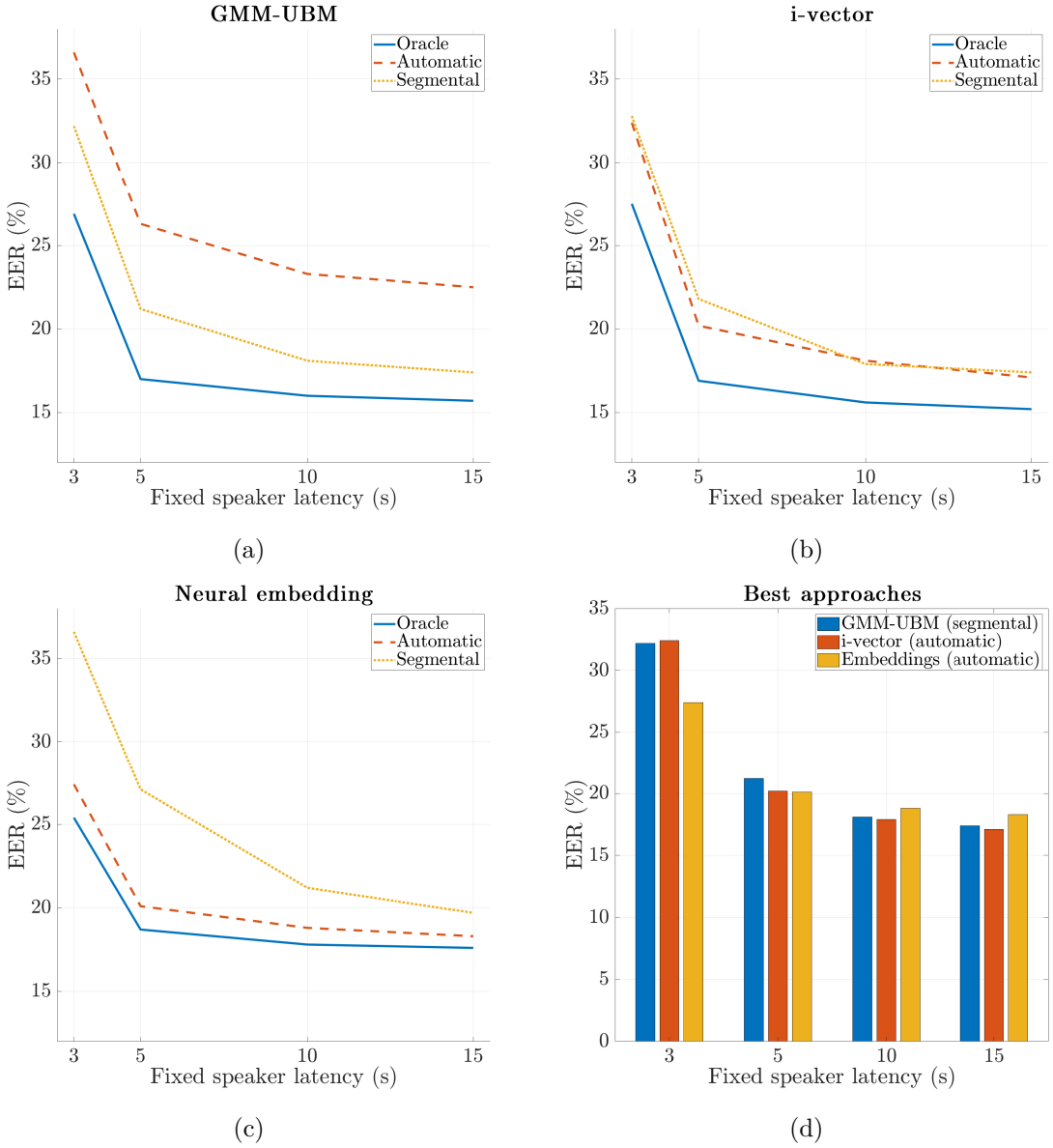


Figure 8.4: Influence of the detection latency on the detection performance on the evaluation set.

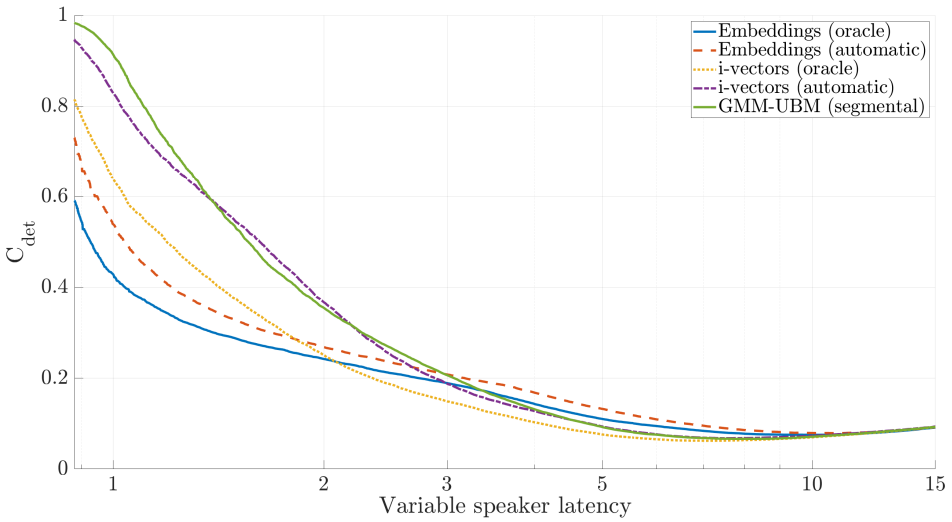


Figure 8.5: Detection performance as a function of the average speaker latency for the best performing automatic systems for the evaluation set. Factors to calculate the C_{det} are those usual of NIST speaker recognition evaluations (SRE) [134], with a $C_{miss} = 10$, $C_{fa} = 1$, and $P_{target} = 0.01$.

GMM-UBM (Fig. 8.4a), i-vector (Fig. 8.4b) and neural embeddings (Fig. 8.4c) speaker detection solutions following either oracle (solid blue line), automatic (dashed orange line) or segmental (thin-dashed yellow line) diarization systems. Finally, Figure 8.4d compares the EER against fixed speaker latency for the best combination of diarization and detection approaches.

No matter what the detection system the best performance is observed with oracle diarization. The performance observed for automatic and segmental diarization systems is dependent upon the detection system. For the GMM system, automatic diarization fares poorly whereas for the neural embedding solution, results for automatic diarization are closer to those obtained with oracle diarization.

While segmental diarization gives reasonable performance in the case of the GMM and i-vector detection systems, performance is poor for the neural embedding detection system. Discrepancies between performance for oracle, automatic and segmental diarization systems are, however, dependent to some degree on the fixed speaker latency, especially for the neural embedding detection system. While differences in performance for oracle and segmental diarization are pronounced for lower fixed latencies, these diminish almost entirely for higher fixed latencies. This diminished effect is however expectable when considering the distribution of amount of target speech in the target trials illustrated in Figure 8.3, which shows that a majority amount for less than 15 seconds of target speech.

Of particular interest is how latency impacts upon the performance of each LLSS solutions and then, which system performs best. A summary of the three first plots of Figure 8.4 for the two practical approaches to diarization (segmental or automatic only - the performance of the oracle system is discounted) is illustrated in the Fig. 8.4d. It shows that the neural embedding detector outperforms the GMM and i-vector systems by a significant margin for the lowest fixed latency of 3s. For higher fixed latencies, however, the GMM and i-vector systems outperform neural embeddings, albeit by a smaller margin; there is little to choose between them.

8.6.2 LLSS performance: variable latency

An illustration of system performance in terms of C_{det} is depicted in Fig. 8.5 for *variable speaker latency* (earlier part of Section 8.3.3). Plots are again illustrated for each detection system and for the best corresponding diarization system (segmental or automatic). Oracle diarization based systems are also plotted for both neural embeddings and i-vectors. C_{det} values are determined according to the usual costs adopted by the NIST speaker recognition evaluations (SRE) [134], i.e. $\mathrm{C}_{miss} = 10$, $\mathrm{C}_{fa} = 1$, and $\mathrm{P}_{target} = 0.01$. Profiles in Fig. 8.5 show a similar picture as that for fixed speaker latencies. Oracle diarization-based systems outperform their automatic counterparts, with i-vectors standing slightly behind in performance against neural embeddings for average latencies lower than 2 seconds, but outperforming them for increasing amounts. Alternatively, the automatic online diarization system and neural embeddings detector shows here too a better performance for lower latencies. Segmental and automatic diarization systems with GMM and i-vector detection systems show lower C_{det} between approximately 3s and 10s, but differences are marginal.

In an alternative interpretation, for a given detection cost, the neural embeddings system provides shorter speaker detection latencies than GMM and i-vector systems. Obviously, selecting the system with minimal C_{det} is not necessarily a sensible strategy for a LLSS task; instead one needs to strike a balance between performance and latency constraints, e.g. selecting the lowest average latency for an admissible cost.

8.6.3 Diarization influences

A first observation on diarization influence is that regarding performance obtained by systems using an oracle online SD clustering algorithm. Looking at all three different detection systems (Figures 8.4a, 8.4b, and 8.4c). The superior performance of this approach over that of automatic and segmental approaches validates the primary premise explored here: correct segmentation and accumulation of homogeneous speaker content

enhances the performance in speaker recognition tasks, even for very low amounts of speech as the ones reported here.

However, performance decreases as soon as an automatic approach is applied to online diarization. Discrepancies in performance between segmental and automatic diarization hypotheses were initially rather puzzling. For the GMM detection system, automatic diarization performs poorly. In contrast, for the neural embedding system, automatic diarization leads to performance that is closer to that of oracle diarization.

The automatic diarization system uses a form of greedy sequential clustering. When performed in an online fashion, all such systems have potential to introduce errors into the diarization hypothesis, errors from which the system can never recover. Impure clusters that contain data from more than one speaker are likely to remain impure as online diarization proceeds. Online diarization performance is illustrated in Table 8.2 for the evaluation partition of the AMI database. Note that DERs are naturally higher than those typically reported in the literature - those reported here relate to an *online* task. Even so, the purity of clusters it produces is reasonable, with over 70% of clusters corresponding to data from the dominant speaker. Coverage, which refers to the percentage of encountered speaker data that is assigned to the corresponding speaker model, exceeds an encouraging 80%.

It is evident that the proposed LLSS systems have different capacities to accommodate errors in the diarization hypothesis. This is mostly due to the different data demands and normalisation strategies employed by each detection solution. Referring to Figure 8.4c, the neural embedding system copes well with data impurities. The GMM and i-vector detection systems cope less well with the same data impurities (the gap between performance for oracle and automatic diarization is greater), however the i-vector system outperforms the neural embedding system for higher latencies by a slight but consistent margin.

In contrast to the automatic diarization system, the segmental approach does not accumulate speaker data through clustering. Diarization performance for the segmental approach is also shown in Table 8.2. While results show a very high DER of 95%, they show that, as expected, purity is higher, while coverage is naturally very low. Thus, while speaker models will be comparatively poorly trained using only short segments of speech, they may yet give better performance in the case that they are trained, more often than not, using data from a single speaker. This is to be expected for such short segments since the chances of them bridging speaker turns is low. As a result, it is not necessarily surprising that the segmental diarization system performs well under some conditions. Eventually, and by pure chance, the detection system will be presented with

Diarization system	DER	Purity	Coverage
Segmental	95.83	88.69	5.73
Automatic	34.24	75.48	81.52

Table 8.2: Online diarization performance in the form of DER (%), cluster purity (%) and coverage (%), obtained with the i-vector automatic online diarization system on evaluation sets, evaluated using a using a standard collar of 250ms.

a pure target speaker segment that will produce a high detection score.

It is clear from the analyses presented above that the dependence of detection systems upon diarization is more complex than may first appear. While the slight difference between automatic and segmental approaches may seem discouraging, the fact that an oracle SD approach significantly outperforms all other approaches guarantees that an improved online clustering algorithm may lead to benefit in performance. At the same time, the small gain between the automatic and segmental approaches reported here *must* be biased by the automatic online clustering algorithm employed for diarization. Similarly, the use of a database that was not explicitly designed for this task is acknowledged as being suboptimal. The protocols presented in Table 8.1 present a only modest speaker variability. These drawbacks, unavoidable in the absence of a purposely-designed dataset, do not overshadow the proposed framework of evaluation for LLSS tasks with the joint operation of online diarization and speaker verification.

Future work should further study the dependence of diarization optimisation towards speaker detection systems, and examine more carefully the robustness to each detection solution to speaker cluster impurities. This may help to better tune the combination of online diarization and speaker detection, thus improving the reliability of LLSS solutions. Results presented in this chapter may suggest that the optimisation of diarization systems with respect to the DER may not be sensible when diarization is only an enabling technology, instead of the final application. Finally, an ideal scenario would be that of collecting and designing a new database designed specifically for research in LLSS. It should naturally contain larger number of speakers and greater speaker variability.

8.7 Summary

This chapter describes a new task termed low-latency speaker spotting (LLSS). The LLSS task is motivated by security and intelligence applications, but has application elsewhere, e.g. voice-based personal assistants, speaker-dependent, but text-independent wake-up systems, and to support further research in short-duration speaker recognition.

The LLSS task calls for the recognition of known, target speakers as quickly as possible after they become active in an audio stream.

Results show that reliable online diarization is key to minimising latency and LLSS performance overall. Differences in results obtained with oracle segmentation and segmental diarization demonstrate the challenge of automatic, online diarization; it can be difficult to outperform a simple segmental approach. Results also show differences in how speaker detection approaches cope with speaker model impurities. Together, these findings show that effective solutions to the LLSS task require a careful combination and joint optimisation of online SD and speaker detection algorithms. They also question the sense of optimising SD, online or otherwise, in isolation when diarization is only an enabling technology, instead of the end application.

Future work should investigate the differences in the behaviour of the proposed speaker detection techniques in detail. It may also investigate strategies to cope with overlapping speech from competing speakers and study more closely combined, joint optimisation of the feature extraction, online diarization and automatic speaker verification components. Emerging end-to-end approaches thus offer another avenue for future work.

Chapter 9

Selective cluster enrichment

Introduced in Chapter 8, low-latency speaker spotting (LLSS) calls for the rapid detection of known speakers within multi-speaker audio streams. While the previous work showed the potential to develop efficient LLSS solutions by combining speaker diarization and speaker detection within an online processing framework, it failed to move significantly beyond the traditional definition of diarization. This chapter shows that the latter needs rethinking and that a diarization sub-system tailored to the end application, rather than to the minimisation of the diarization error rate, can improve LLSS performance. The work presented here introduces a selective cluster enrichment (SCE) algorithm used to guide the diarization system to better model segments within a multi-speaker audio stream and hence detect more reliably a *given target speaker*. The LLSS solution reported in this chapter shows that target speakers can be detected with a 15.86% equal error rate after having been active in online-processed multi-speaker audio streams for only 15 seconds, achieving 10% relative improvement over the results reported in Chapter 8. The work and results reported here were published in [249].

The remainder of this chapter is organised as follows. Section 9.1 describes motivations in the context of the LLSS framework reported in Chapter 8 [226]. Section 9.2 describes the new SCE procedure. Experiments are reported in Section 9.3. A summary is given in Section 9.4.

9.1 Introduction

Our first attempt to develop an efficient LLSS solution [226] took the first steps to unite the optimisation of speaker detection and speaker diarization technologies within a common online framework. While that work showed the potential, it failed to move

significantly beyond the traditional definition of diarization. This work aims to redefine the diarization problem such that the solution is more closely married to the core LLSS task. The approach exploits the use of the target speaker model (the one which is pre-trained for the speaker detection task) to guide diarization to cluster more reliably matching segments in the incoming audio stream. The process is referred to as SCE.

9.2 Selective cluster enrichment

This section describes the adaptation of a diarization sub-system to the operation of a subsequent speaker detection sub-system.

Speaker diarization systems typically entail some form of segmentation and clustering process in order to determine the number of speakers within a multi-speaker audio stream and *who speaks when*. Generally, diarization is performed offline, meaning a diarization algorithm has access to the full audio stream before deriving a diarization hypothesis. In contrast, online diarization can be performed by processing an audio stream in segmental or sequential fashion and by updating the current diarization hypothesis to account for new speech data as it is encountered.

Be them offline or online, speaker diarization systems are usually evaluated using the classical DER which combines measures of background noise mistaken for speech, speech mistaken for background noise and speech assigned to the wrong speaker. In practice, one must strike a balance between under and over clustering. When the number of clusters is too high, i.e. greater than the number of speakers, then resulting clusters may have high purity – they are not contaminated excessively by the data of other speakers – by resulting models tend to be poorly trained using insufficient data [100]. In contrast, when the number of clusters is too few, models are comparatively well trained using more data, but purity decreases – inhomogeneous clusters are trained using data from multiple speakers. Somewhere in between, the balance between data quantity and impurity helps to minimise the DER or, as is the goal of the work reported in this chapter, to optimise a more application-inspired metric.

The research hypothesis under investigation in the work reported here is that there is potential to guide the clustering process in a way that better balances data quantity and purity in order to improve the reliabilty of a subsequent speaker detection algorithm. This idea is explored within the context of a LLSS task which seeks to detect a particular target speaker for which a model is already trained and available. It seems logical in this case that the diarization process should at least make use of the target speaker model.

The original LLSS approach uses an online speaker diarization process that produces

an evolving diarization hypothesis comprising n clusters c_1^t ... c_n^t. Newly arriving data is assigned to the closest cluster in the current diarization hypothesis. The set of clusters are then scored against the target speaker models giving a set of scores s_1^t ... s_n^t. The maximum score among them is then compared to threshold θ in order to derive the detection decision $\Gamma(t)$.

The proposed modification is illustrated in Figure 9.1. The idea is to consider the target speaker model in the assignment of newly arriving data to one of the clusters in the current diarization hypothesis. The closest matching cluster is derived as before. In contrast to the original approach, though, the newly trained cluster for time t is replaced by the previous cluster for time $t-1$ if the new cluster score s_n^t is less than the previous cluster score s_n^{t-1}. The result is that the closest matching cluster is enriched with newly arriving data only if it improves the match between the cluster and the target speaker model. According to the max operation to the right of Figure 9.1, the set of scores s_1^t ... s_n^t will then be monotonically increasing with t. As before, the largest of these scores is then compared to threshold θ in order to derive the detection decision $\Gamma(t)$.

Even if the use of the target model at the heart of the diarization process is entirely intuitive, the motivation for the specific way in which it is used is far less intuitive. We attempt now to explain why its use in this way should lead to better LLSS performance. Selective cluster enrichment will have one of two effects. In the case that the closest cluster to newly arriving data match well the target model, then the process will serve only to purify the cluster, increase still further the match with the target model and improve LLSS performance. Other clusters that do not match well the target model can still only be enriched or adapted towards the target speaker model. In the case that the audio stream does not contain speech from the target speaker, then clusters will be either poorly trained using very little data, in which case diarization performance will deteriorate, or they will be adapted successfully towards the target, thereby degrading LLSS performance (since the speakers do not match). The hypothesis is that, even if clusters are inadvertently adapted to the target model, they will rarely be adapted sufficiently well such that the likelihood exceeds the detection threshold. In this case, the benefit of purifying matching clusters will outweigh the penalty of inadvertently adapting non-matching clusters. Accordingly, SCE should help to improve LLSS performance.

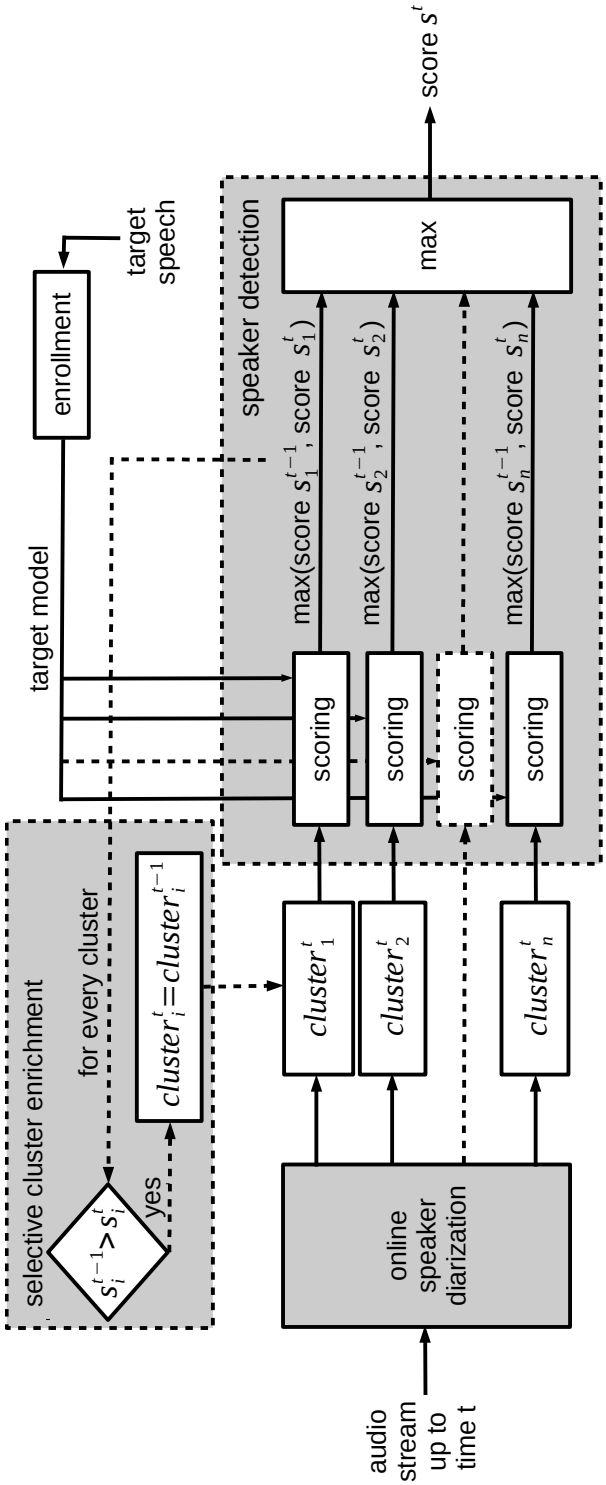


Figure 9.1: An illustration of the low-latency speaker spotting solution that combines online speaker diarization with detection. The baseline LLSS pipeline illustrated in Figure 8.2 is adapted to incorporate SCE.

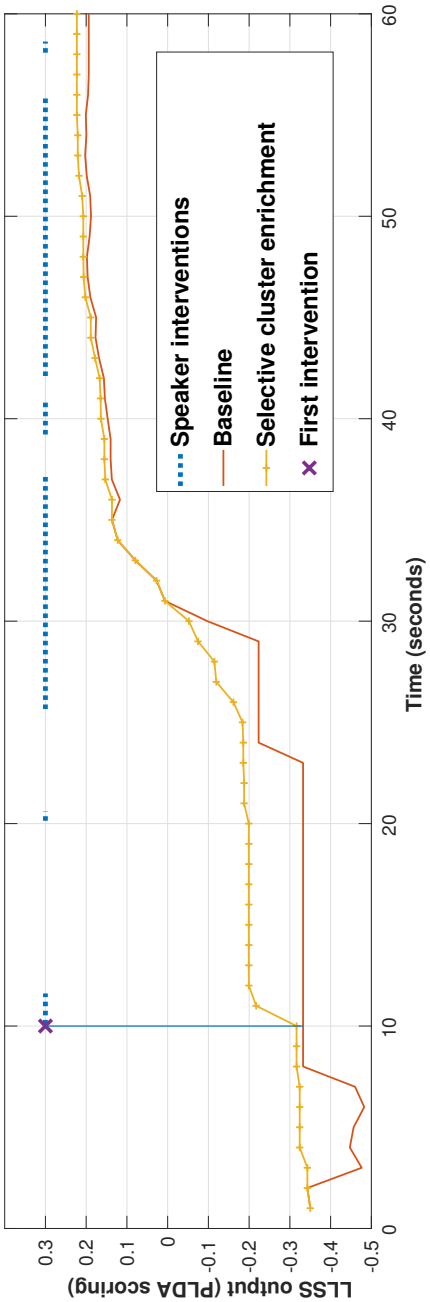


Figure 9.2: An illustration of LLSS using PLDA scoring for an arbitrary trial utterance containing a target speaker. Profiles shown for the baseline and proposed solution with SCE.

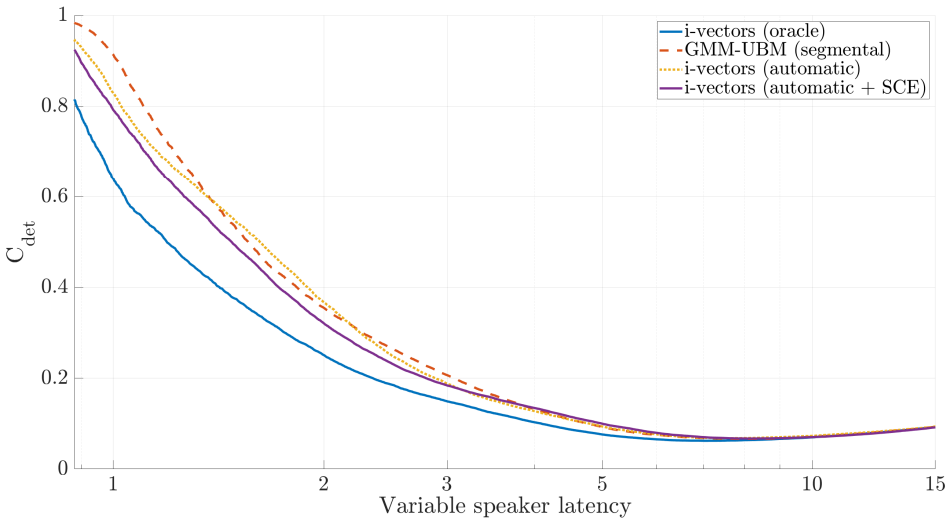


Figure 9.3: Detection performance as a function of the average speaker latency for the best performing automatic systems on the evaluation set. The application of the proposed SCE approach benefits the i-vector automatic system for very low values of speaker latency.

9.3 Experimental work

This section describes experiments designed to evaluate LLSS performance and the benefit of SCE.

9.3.1 General setup

The database used for all experimental work reported here, as well as the protocol tested is exactly the same as that used in the original LLSS work in Chapter 8 [226]. Differences are present in that of the speaker detection systems employed. In Chapter 8 performance is reported for GMM-UBM, i-vector, and neural embeddings systems. Results for the latter are not reported here since this work was undertaken independently to ODESSA partners LIMSI who provided the neural embeddings solely used for work reported in Chapter 8.

9.3.2 Results

Results in Table 9.1 illustrate LLSS performance for the baseline and proposed solution, for both oracle and automatic diarization and for both GMM-UBM and i-vector speaker detection algorithms. Performance is expressed in terms of the equal error rate (EER) for various fixed speaker latencies: 3, 5, 10 and 15 seconds. Similarly, performance as

		GMM-UBM				i-vector			
	Latency	3	5	10	15	3	5	10	15
Baseline	Oracle	26.9	17	15.99	15.69	27.49	16.94	15.56	15.24
	Segmental	32.2	21.2	18.1	17.4	32.8	21.8	17.09	17.04
	Automatic	36.56	26.3	23.28	22.53	32.41	20.2	18.06	17.31
Proposed	Oracle	27.77	17.81	16.32	15.87	29.34	18.33	15.95	15.33
	Automatic	34.32	23.86	20.98	20.22	**31.08**	**19.27**	**16.61**	**15.82**

Table 9.1: LLSS performance illustrated in terms of EER for different fixed speaker latencies.

a function of the C_{det} (defined as in previous chapter and using the same NIST SRE parameters), is illustrated in Figure 9.3 for variable amounts of average speaker latency. On account of non-target models being poorly trained, results show that performance universally degrades for oracle diarization. However, results in Table 9.1 universally improve in the case of automatic diarization. This is due to improvements to target model purity stemming from SCE. While it is not the goal of this work to compare GMM-UBM and i-vector algorithms, it is reassuring that SCE improves the performance of both.

Similar tendencies are observed in terms of variable speaker latency, for which results are shown in Figure 9.3. The C_{det} obtained by an i-vector based oracle diarization (solid blue line) represents the best achievable performance. Results for an automatic diarization i-vector system (dotted yellow line) and a GMM-UBM segmental approach (dashed orange line) constitute the baseline performance as reported in Chapter 8. These baseline results show that an automatic diarization system struggles to outperform a simpler segmental-based approach. This may be explained by the lower cluster purity of 75% (34% DER) for the i-vector system with regard to the 88% cluster purity (and 95% DER) for the GMM-UBM segmental approach. The proposed approach manages to overcome the errors in the online clustering algorithm bringing the performance of the resulting system (solid purple line) closer to that of the oracle system. These results show the positive impact to ASV performance of the proposed algorithm and confirm the research hypothesis. Closer integration of diarization and recognition is beneficial. Furthermore, better performance can be achieved, even when diarization performance (when assessed independently) is worse.

Further evidence is illustrated in Figure 9.2 which shows the evolution in PLDA scoring for the baseline and proposed LLSS solutions for an arbitrary utterance that contains the target speaker during the intervals indicated towards the top of the plot. The LLSS output for the proposed system is consistently higher than that of the baseline, showing that SCE serves to improve purity, forcing a monotonic increase in the score.

9.4 Summary

This chapter shows how the performance of a low-latency speaker spotting (LLSS) solution can be improved by tailoring the operation of a speaker diarization sub-system to that of the following speaker detection sub-system. The proposed SCE scheme exploits the target speaker model to guide the diarization process in order to enhance the purity of matching clusters in the diarization hypothesis. The work serves to show that the optimisation of a diarization system on its own will never produce optimal results when diarization is only but one component of a more complex toolchain. Selective cluster enrichment will surely degrade the reliability of the diarization hypothesis when assessed with the traditional diarization error rate, but it nonetheless leads to more reliable speaker detection and LLSS performance. Universal improvements observed across two different speaker detection algorithms and a range of different speaker latencies show the potential for still further improvements using more elaborate end-to-end optimisations.

Chapter 10

Conclusions

This chapter provides a summary of the results and findings from the work reported in this thesis. This material is reported in Section 10.1. It also presents some directions for future research are presented in Section 10.2.

10.1 Summary

The topic central to this thesis is speaker diarization (SD), an important pre-processing tool for the tasks of conversational analysis, content indexing, or, as regards this thesis, speaker recognition. The research probed two different angles. First, it reports a study of domain mismatch leading to the training-data independent, domain-robust approach to SD based on binary key (BK) speaker modelling [105]. Second, it reports a study of a practical application of SD concerning the rapid detection of blacklisted speakers. In contrast to most work in SD, it concerns the *joint* assessment of online SD and automatic speaker verification (ASV).

Chapter 4 proposes an alternative to traditional spectral analysis as a means of improving the discriminative capacity of BK speaker modelling. Traditional acoustic features (Mel-frequency cepstral coefficients (MFCCs)) are derived from spectral analysis based on the short-time Fourier transform (STFT). Here, MFCCs are compared to the recently developed infinite-impulse response, constant Q transform (IIR-CQT) Mel-frequency cepstral (ICMC) coefficients [160] in their first application to BK speaker modelling. Given the novel front-end, the work also analyses the impact of frame lengths that exceed those typically used for short-term feature extraction. Experiments in terms of controlled ASV and fully fledged diarization show that: the discriminative capacity of BK speaker modelling benefits from increased frame lengths for both evaluated front-

end techniques; the multi-resolution nature of ICMC features and IIR-CQT spectral analysis outperforms STFT-based MFCCs, in both speaker recognition and diarization. These findings are confirmed through other work reported later in the thesis, specifically in Chapter 6, in which the same front-ends are compared using a different dataset. The enhanced front-end for BK-based SD was also compared to that of other systems submitted to the Albayzin 2016 Speaker Diarization Evaluation, in which the submitted system obtained 1st place.

The problem of explicit speaker change detection (SCD), of use for SD and other related applications, had never been addressed by means of BK speaker modelling before the undertaking of this thesis. The baseline system defined in Chapter 3 performs *implicit* SCD through the segmentation of speech in chunks of short length. The work reported in **Chapter 5** investigates *explicit* SCD. The potential is assessed through two different methods of composing the binary key background model (KBM). The traditional baseline approach is compared to a second, novel method that attempts to better exploit local contextual information in the speech content surrounding hypothesised speaker change points. Results in terms of BK-based SCD are positive, showing consistently better results than the baseline system based on the Bayesian information criterion (BIC). The work considers two segment-level representations derived from the KBM, i.e. cumulative vectors (CVs) and binary keys (BKs). BKs are found to obtain better speaker segmentation performance than CVs. The difference is attributed to the more aggressive scoring method between BKs using the Jaccard distance.

In keeping with other, similar work on the same topic, SCD experiments were performed using the ETAPE databse. The proposed training-free BK approach is shown to perform competitively compared to deep learning (DL) inspired approaches, though without the need for domain-matched training data. The proposed SCD approach is also assessed with regard to its influence on SD performance. Different methods to SCD are applied to the baseline BK-based SD system. These include the baseline homogeneous segmentation, a splitting of the speech content based on the boundaries of the voice activity detection (VAD) system, the explicit BK-based SCD approach, and their combination. Results in terms of diarization performance show the benefit of the proposed method when coupled with a baseline MFCC-based acoustic front-end. However, when tested using the better-performing ICMC features (Chapter 4), the benefit of explicit SCD diminishes. While slightly discouraging, this finding corroborates those of other works [56, 59] which show inconsistent benefit of SCD to SD performance. Even so, SCD may still be beneficial for other, related applications for which the efficiency of BK modelling may still be an advantage in terms of competing techniques.

Chapter 6 relates to clustering. This work is reported in the context of the first

DIHARD challenge. The DIHARD dataset includes much more diverse data domains than those usually considered in the SD literature and other evaluations. This diversity presents a significant challenge for SD technology. Work reported in Chapter 6 attempted to tackle domain robustness using a BK-based SD system. The solution is to avoid domain-specific training data entirely, while remaining competitive with the systems submitted by leading international research institutions.

An analysis of the baseline system, enhanced by means of the acoustic front-end proposed in Chapter 4, showed that the baseline agglomerative hierarchical clustering (AHC) algorithm is responsible for poor performance. Particularly, the erroneous selection of the number of speakers was most harmful to performance. This observation motivated the search for alternative methods. For the first time, spectral clustering (SC) was applied and optimised for its application in a BK-based diarization solution using CVs. Affinity matrices, calculated for in-session CVs and necessary for the eigendecomposition involved in SC, are refined by a number of pattern-enhancing operations designed to improve speaker discrimination. SC proved to be a reliable, domain-robust alternative to clustering for BK-based diarization systems. In order to estimate the number of speakers, the maximum eigengap between eigenvalues was used as a criterion. This work resulted in a more robust approach to estimate the number of speakers. When integrated into the baseline AHC system, use of the new approach to estimate the number of speakers produced the 2nd most significant improvement to the baseline system. Furthermore, the novel approach to single speaker detection which is based on the distance between eigenvalues leads to additional improvements in performance. Compared to leading alternatives, the resulting BK-based diarization system obtains competitive results. These are close to those of other systems for which results were also submitted to the DIHARD challenge. These are mostly based on data-hungry and computationally demanding DL solutions. While the difference in performance is modest, the BK-based SD system requires no external training data, is thus readily applicable to data in any domain and is significantly less computationally demanding.

The final contribution to the development of BK-based SD, reported in **Chapter 7**, took a slightly different direction. It is argued in this thesis that independence to training data is the main asset of the BK approach to SD which makes it inherently domain insensitive. Deep learning approaches in the form of speaker embeddings are nonetheless slightly superior in performance when it comes to test sets whose domains match those available for training. In consequence, and to investigate the complementary character of training dependent and independent methods to SD, the work reported in Chapter 7 describes the first effort to link BK-based solutions with neural embedding-based approaches. The Albayzin 2018 Speaker Diarization Challenge provided a new dataset and two training conditions with which to explore system combination/fusion.

The method to SD fusion applied on the open-set condition of the Albayzin 2018 database is based upon the integration of time-aligned systems via a similarity-matrix approach. This kind of operation can be detrimental to the performance of the individual systems. However, open-set conditions to imply that neural embeddings may leverage large amounts of external training data. The speaker modelling techniques used within this fusion approach involve BK-based CVs, x-vectors and triplet-loss neural embedddings [57]. The resulting fused system earned 1st place for the open-set training condition. The fusion approach used for the closed-set training condition involved BK and triplet-loss neural embedding-based systems. Use of x-vectors was discarded as performance was poor when trained with constrained data. Fusion was performed at the hypothesis level (where synchrony between systems is not necessary), and is based upon a label-merging technique. The resulting system obtained 2nd place in the Albayzin 2018 Speaker Diarization Challenge.

Findings from the first line of research are summarised as follows:

- Use of front-ends with larger frame duration lead to a relative improvement in diarization error rate (DER) of a 9% for MFCCs. The replacement of STFT spectral analysis by IIR-CQT-based ICMC coefficients lead to a relative improvement of 14% DER.
- Use of BK-based CVs for explicit SCD leads to a relative increase in average coverage of 10% over the BIC-based baseline for both KBM-composition methods. Use of BKs results in a relative improvement of 17.4% for a global-context KBM. The novel, local-context approach to the KBM results in a 18.3% increase.
- Use of spectral clustering (SC) leads to relative improvements in DER over the baseline system of (i) 30% when SC is used to estimate the number of speakers and clustering, (ii) 37% when also coupled with the baseline AHC, and (iii) 40% when also used for single-speaker detection.
- Using a similarity-matrix, fusion of BK and DL based SD solutions leads to an 8% relative improvement in DER for the open-set condition of the Albayzin 2018 Speaker Diarization Challenge. For the closed-set training condition and a hypothesis-level fusion, a 13% relative improvement was achieved.

The second line of research in this manuscript investigates SD in a real use case. EURECOM's participation in the joint-national ODESSA project helped define this line of work. Its goal concerns the development of solutions for the rapid, online detection of previously enrolled speakers in multi-speaker audio streams. Such research addresses

two limitations of current ASV technology, namely (i) robust speaker detection using only short test utterances and (ii) robust speaker detection in the presence of multiple speakers. The new task is coined as low-latency speaker spotting (LLSS).

Chapter 8 defines the task and proposes a framework for its investigation. Contributions include: (i) the inclusion of speaker latency into ASV assessment; (ii) a protocol that supports LLSS research using a public database; (iii) the development of an i-vector based online SD system; (iv) the benchmarking of three different LLSS solutions employing GMM-UBM, i-vectors, and neural embeddings techniques. Results highlight the challenging of the proposed LLSS task which stems from the difficulty in clustering in online fashion using very small quantities of data. As a result, there is only modest difference in the performance of each approach to ASV. SD is thus the bottleneck and better solutions to online clustering are needed.

An attempt to the improve LLSS performance is presented in **Chapter 9**. The work aims to fuse online diarization and speaker detection systems at their heart and with the final application in mind from the start. This work marks a departure from traditional speaker diarization research.

This approach, named selective cluster enrichment (SCE) serves to guide the online sequential clustering algorithm. This is achieved by leveraging the target speaker model, usually used only for speaker detection, for clustering in the SD stage. SCE acts to purify the cluster which corresponds to the *target* speaker. While it also acts to minimise the effect to detection of other speaker models resulting from clustering. Results are reported for GMM-UBM and i-vector approaches to ASV and show the benefit of the approach. Of particular interest is the degradation to SD performance which nonetheless results in better speaker detection performance. The results show that optimisation of an SD solution when it is not the final application is not necessarily sensible.

The findings from the second line of research are summarised as follows:

- The online SD system based on the sequential clustering of i-vectors delivers a performance of 34 % DER and associated cluster purity/coverage of 75/82%. The segmental, non-clustered approach to SD delivers a comparatively higher DER of 96% and a considerably lower coverage of 6%, but a higher purity of nearly 89%.

- Even though it degrades SD performance, SCE improves LLSS performance. Compared to the baseline system, the EER is reduced by a 4% relative for speaker latencies of 3 and 5s, 8% for 10s, and a 9% for 15s.

10.2 Directions for future research

Directions for future research relate to both further study of the BK-based approach to SD as well as continued research in LLSS. Further work as regards the BK-based approach to SD include:

- **Data augmentation:** Recent developments in DL solutions to speaker modelling [5] were made possible through (i) the use of topologies that led to more discriminative speaker modelling, but also, very importantly, (ii) the exploitation of huge amounts of data, including acoustically modified versions of the training data that increases data quantity and also data variability. Its independence from external training data is an asset of BK-based approaches to SD. Even so, the KBM is still trained, albeit with test data. Data augmentation, by means of added noise and/or reverberation, but applied at test time (when the KBM is learned) could help to better model in-session variability Further work should thus study the coupling of data augmentation techniques with BK-based modelling. It is stressed that such work would remain independent from *external* training data.
- **Domain adaptation:** BK-based approaches to SD have relatively few *tunable* hyper-parameters. However, KBM parameters can still be tuned to deliver better results for some domain-specific scenarios. Training data-dependent methods to speaker modelling usually rely on hyper-parameters tuned over an extensive training set in order to learn a *general* representation of the speaker space. Here though, a generic set of parameters could be undesirable, especially given that the KBM is a strictly *local* representation of the acoustic space. In this sense, the development of KBM training techniques that adapt better and more specifically to the domain may have scope to improve domain robustness further.
- **Better fusion with DL approaches:** The work presented in this thesis shows that BK-based approaches to SD compare favourably to the more data-hungry and computationally demanding techniques. This makes BK-based solutions appealing when suitable training data is scarce even if performance can fall slightly short of that of competing systems, there is potential benefit of combining BK-based systems with DL based pipelines. Future work should investigate further the fusion of SD systems and especially those which cope with asynchronous label boundaries.

Ideas for future research related to LLSS include:

- **LLSS-oriented dataset collection:** The Augmented Meeting Interaction (AMI) database [248] used for the work in LLSS reported in this thesis was not collected

with speaker recognition let alone LLSS tasks in mind. The collection of an LLSS-oriented database would be essential to future work. Such a database would need transcriptions similar to those used in SD research. The same databases could also be used for research in short duration ASV.

- **Online target speaker extraction:** Much research has appeared recently in the literature relating to the problem of target speaker extraction and separation in the context of overlapping, multi-speaker speech [250, 251, 252, 253, 254]. Most of these methods generate speaker-dependent acoustic masks applicable at the spectrogram level to separate speech. This kind of technology could be of benefit to the LLSS task by separating mixed speech in the occasional presence of speaker overlap. Ideally, separate threads of speech could be generated for all participating speakers, potentially benefiting both diarization and LLSS performance. Future work should consider such approaches and online implementations suited to both tasks.

Bibliography

[1] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," *IEEE Transactions on audio, speech, and language processing*, vol. 14, no. 5, pp. 1557–1565, 2006. [Cited on pages 2, 15].

[2] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker Diarization: A Review of Recent Research," *IEEE Trans. on Audio, Speech, and Language Proc.*, vol. 20, no. 2, pp. 356–370, Feb. 2012. [Cited on pages 2, 15, 131].

[3] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech communication*, vol. 52, no. 1, pp. 12–40, 2010. [Cited on pages 2, 15, 16, 17].

[4] J. H. Hansen and T. Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal processing magazine*, vol. 32, no. 6, pp. 74–99, 2015. [Cited on pages 2, 15, 28, 130].

[5] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018. [Cited on pages 3, 24, 158].

[6] M. Rouvier, P.-M. Bousquet, and B. Favre, "Speaker diarization through speaker embeddings," in *Proc. European Signal Processing Conference (EUSIPCO)*, 2015, pp. 2082–2086. [Cited on pages 3, 24, 89].

[7] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 4930–4934. [Cited on pages 3, 24, 27, 89].

[8] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. L. Moreno, "Speaker diarization with LSTM," in *Proc. IEEE International Conference on Acoustics,*

Speech and Signal Processing (ICASSP), 2018, pp. 5239–5243. [Cited on pages 3, 24, 26, 89, 95, 96, 97].

[9] X. Anguera and J.-F. Bonastre, "A novel speaker binary key derived from anchor models," in *Proc. INTERSPEECH*, 2010, pp. 2118–2121. [Cited on pages 3, 35, 36, 39, 41, 46].

[10] ——, "Fast speaker diarization based on binary keys," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 4428–4431. [Cited on pages 3, 41, 50].

[11] H. Delgado, X. Anguera, C. Fredouille, and J. Serrano, "Fast single-and cross-show speaker diarization using binary key speaker modeling," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 23, no. 12, pp. 2286–2297, 2015. [Cited on pages 3, 71, 88, 89].

[12] A. Ortega, I. Viñals, A. Miguel, and E. Lleida, "The Albayzin 2016 speaker diarization evaluation," in *Proc. IberSPEECH*, 2016. [Cited on pages xii, 5, 51, 56, 57, 88, 89, 116].

[13] A. Ortega, I. Viñals, A. Miguel, E. Lleida, V. Bazán, C. Pérez, M. Zotano, and A. de Prada, "Albayzin Evaluation: IberSPEECH-RTVE 2018 Speaker Diarization Challenge," 2018. [Cited on pages 5, 88, 89, 110].

[14] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, "First DIHARD challenge evaluation plan," 2018. [Cited on pages 5, 88, 90, 101, 102].

[15] C. Vaquero, O. Vinyals, and G. Friedland, "A hybrid approach to online speaker diarization," in *Proc. INTERSPEECH*, 2010. [Cited on pages 7, 26].

[16] J. Geiger, F. Wallhoff, and G. Rigoll, "GMM-UBM based open-set online speaker diarization," in *Proc. INTERSPEECH*, 2010. [Cited on pages 7, 26].

[17] G. Soldi, C. Beaugeant, and N. Evans, "Adaptive and online speaker diarization for meeting data," in *Proc. European Signal Processing Conference (EUSIPCO)*, 2015, pp. 2112–2116. [Cited on pages 7, 26].

[18] P. Rose, *Forensic speaker identification.* cRc Press, 2002. [Cited on page 15].

[19] S. B. Davis and P. Mermelstein, "Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980. [Cited on pages 15, 111].

[20] J. Makhoul, "Linear prediction: A tutorial review," *Proceedings of the IEEE*, vol. 63, no. 4, pp. 561–580, 1975. [Cited on page 15].

[21] X. Huang, A. Acero, H.-W. Hon, and R. Foreword By-Reddy, *Spoken language processing: A guide to theory, algorithm, and system development*. Prentice hall PTR, 2001. [Cited on page 15].

[22] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *the Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, 1990. [Cited on page 15].

[23] G. Friedland, O. Vinyals, Y. Huang, and C. Muller, "Prosodic and other long-term features for speaker diarization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 5, pp. 985–993, 2009. [Cited on page 16].

[24] A. W. Zewoudie, J. Luque, and J. Hernando, "The use of long-term features for gmm-and i-vector-based speaker diarization systems," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2018, no. 1, p. 14, 2018. [Cited on page 16].

[25] J. Kreiman and B. R. Gerratt, "Perception of aperiodicity in pathological voice," *The Journal of the Acoustical Society of America*, vol. 117, no. 4, pp. 2201–2211, 2005. [Cited on page 16].

[26] X. Li, J. Tao, M. T. Johnson, J. Soltis, A. Savage, K. M. Leong, and J. D. Newman, "Stress and emotion classification using jitter and shimmer features," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2007, pp. IV–1081. [Cited on page 16].

[27] G. Doddington, "Speaker recognition based on idiolectal differences between speakers," in *Proc. INTERSPEECH*, 2001. [Cited on page 16].

[28] M. Li, L. Liu, W. Cai, and W. Liu, "Generalized i-vector representation with phonetic tokenizations and tandem features for both text independent and text dependent speaker verification," *Journal of Signal Processing Systems*, vol. 82, no. 2, pp. 207–215, 2016. [Cited on page 16].

[29] Y. Zhang, F. Weninger, B. Liu, M. Schmitt, F. Eyben, and B. Schuller, "A paralinguistic approach to speaker diarisation: using age, gender, voice likability and personality traits," in *Proc. ACM international conference on Multimedia*, 2017, pp. 387–392. [Cited on page 16].

[30] S. Yaman, J. Pelecanos, and R. Sarikaya, "Bottleneck features for speaker recognition," in *Proc. Speaker Odyssey*, 2012. [Cited on page 16].

[31] A. Lozano-Diez, A. Silnova, P. Matejka, O. Glembek, O. Plchot, J. Pesan, L. Burget, and J. Gonzalez-Rodriguez, "Analysis and Optimization of Bottleneck Features for Speaker Recognition," in *Proc. Speaker Odyssey*, 2016, pp. 21–24. [Cited on page 16].

[32] J. Jorrín, P. García, and L. Buera, "DNN bottleneck features for speaker clustering," in *Proc. INTERSPEECH*, 2017, pp. 1024–1028. [Cited on page 16].

[33] S. H. Yella, A. Stolcke, and M. Slaney, "Artificial neural network features for speaker diarization," in *Proc. IEEE Spoken Language Technology Workshop (SLT)*, 2014, pp. 402–406. [Cited on page 16].

[34] S. H. Yella and A. Stolcke, "A comparison of neural network feature transforms for speaker diarization," in *Proc. INTERSPEECH*, 2015. [Cited on page 16].

[35] I. Viñals, J. Villalba, A. Ortega, A. Miguel, and E. Lleida, "Bottleneck based front-end for diarization systems," in *Proc. IberSPEECH*, 2016, pp. 276–286. [Cited on page 16].

[36] X. Anguera, C. Wooters, B. Peskin, and M. Aguiló, "Robust speaker segmentation for meetings: The ICSI-SRI spring 2005 diarization system," in *Proc. International Workshop on Machine Learning for Multimodal Interaction*, 2005, pp. 402–414. [Cited on page 17].

[37] X. Zhu, C. Barras, L. Lamel, and J.-L. Gauvain, "Multi-stage speaker diarization for conference and lecture meetings," in *Multimodal technologies for perception of humans*. Springer, 2007, pp. 533–542. [Cited on page 17].

[38] E. Rentzeperis, A. Stergiou, C. Boukis, A. Pnevmatikakis, and L. C. Polymenakos, "The 2006 Athens Information Technology Speech activity detection and speaker diarization systems," in *Proc. International Workshop on Machine Learning for Multimodal Interaction*, 2006, pp. 385–395. [Cited on page 17].

[39] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967. [Cited on page 17].

[40] X.-L. Zhang and D. Wang, "Boosting contextual information for deep neural network based voice activity detection," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 24, no. 2, pp. 252–264, 2016. [Cited on page 17].

[41] Y. Obuchi, "Framewise speech-nonspeech classification by neural networks for voice activity detection with statistical noise suppression," in *Proc. IEEE International*

Conference on Acoustics, Speech and Signal Processing (ICASSP), 2016, pp. 5715–5719. [Cited on page 17].

[42] S.-Y. Chang, B. Li, G. Simko, T. N. Sainath, A. Tripathi, A. van den Oord, and O. Vinyals, "Temporal modeling using dilated convolution and gating for voice-activity-detection," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5549–5553. [Cited on page 17].

[43] R. Zazo Candil, T. N. Sainath, G. Simko, and C. Parada, "Feature learning with raw-waveform CLDNNs for voice activity detection," in *Proc. INTERSPEECH*, 2016. [Cited on page 17].

[44] G. Gelly and J.-L. Gauvain, "Minimum word error training of RNN-based voice activity detection," in *Proc. INTERSPEECH*, 2015. [Cited on pages 17, 112].

[45] G. Schwarz *et al.*, "Estimating the dimension of a model," *The annals of statistics*, vol. 6, no. 2, pp. 461–464, 1978. [Cited on page 18].

[46] S. Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the bayesian information criterion," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, vol. 8, 1998, pp. 127–132. [Cited on pages 18, 19, 70, 75].

[47] D. Liu and F. Kubala, "Fast speaker change detection for broadcast news transcription and indexing," in *Proc. EUROSPEECH*, 1999. [Cited on page 19].

[48] P. Delacourt, D. Kryze, and C. J. Wellekens, "Speaker-based segmentation for audio data indexing," in *Proc. ESCA Tutorial and Research Workshop (ETRW) on Accessing Information in Spoken Audio*, 1999. [Cited on page 19].

[49] J.-F. Bonastre, P. Delacourt, C. Fredouille, T. Merlin, and C. Wellekens, "A speaker tracking system based on speaker turn detection for NIST evaluation," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, 2000, pp. II1177–II1180. [Cited on page 19].

[50] M. A. Siegler, U. Jain, B. Raj, and R. M. Stern, "Automatic segmentation, classification and clustering of broadcast news audio," in *Proc. DARPA speech recognition workshop*, 1997. [Cited on page 19].

[51] A. S. Malegaonkar, A. M. Ariyaeeinia, and P. Sivakumaran, "Efficient speaker change detection using adapted Gaussian mixture models," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 6, pp. 1859–1869, 2007. [Cited on pages 20, 70].

[52] T.-Y. Wu, L. Lu, K. Chen, and H. Zhang, "Universal Background Models for Real-time Speaker Change Detection," in *Proc. MMM*, 2003, pp. 135–149. [Cited on pages 20, 70].

[53] Z. Zajíc, M. Kunešová, and V. Radová, "Investigation of Segmentation in i-Vector Based Speaker Diarization of Telephone Speech," in *Proc. International Conference on Speech and Computer*. Springer, 2016, pp. 411–418. [Cited on pages 20, 70].

[54] L. V. Neri, H. N. Pinheiro, T. I. Ren, G. D. d. C. Cavalcanti, and A. G. Adami, "Speaker segmentation using i-vector in meetings domain," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5455–5459. [Cited on pages 20, 70].

[55] V. Gupta, "Speaker change point detection using deep neural nets," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4420–4424. [Cited on pages 20, 70, 75, 84, 86].

[56] M. Hrúz and Z. Zajíc, "Convolutional Neural Network for speaker change detection in telephone speaker diarization system," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 4945–4949. [Cited on pages 20, 70, 154].

[57] H. Bredin, "Tristounet: Triplet loss for speaker turn embedding," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5430–5434. [Cited on pages 20, 24, 70, 75, 76, 82, 83, 112, 136, 156].

[58] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, "Learning fine-grained image similarity with deep ranking," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1386–1393. [Cited on pages 20, 112].

[59] R. Yin, H. Bredin, and C. Barras, "Speaker change detection in broadcast TV using bidirectional long short-term memory networks," in *Proc. INTERSPEECH*, 2017. [Cited on pages 20, 75, 82, 83, 86, 112, 135, 154].

[60] D. A. Reynolds, "Speaker identification and verification using gaussian mixture speaker models," *Speech communication*, vol. 17, no. 1-2, pp. 91–108, 1995. [Cited on page 20].

[61] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE Transactions on Audio, Speech and Audio processing*, vol. 3, no. 1, pp. 72–83, 1995. [Cited on page 20].

[62] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital signal processing*, vol. 10, no. 1-3, pp. 19–41, 2000. [Cited on pages 20, 21].

[63] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977. [Cited on page 21].

[64] D. A. Reynolds, "Comparison of background normalization methods for text-independent speaker verification," in *Proc. EUROSPEECH*, 1997. [Cited on page 21].

[65] C. Wooters and M. Huijbregts, "The ICSI RT07s speaker diarization system," in *Multimodal Technologies for Perception of Humans*. Springer, 2007, pp. 509–519. [Cited on page 21].

[66] T. Nguyen, H. Sun, S. Zhao, S. Z. K. Khine, H. D. Tran, T. L. N. Ma, B. Ma, E. S. Chng, and H. Li, "The IIR-NTU speaker diarization systems for RT 2009," in *Proc. RT'09, NIST Rich Transcription Workshop*, vol. 14, 2009, pp. 17–40. [Cited on page 21].

[67] A. Solomonoff, C. Quillen, and W. M. Campbell, "Channel compensation for SVM speaker recognition," in *Proc. Speaker Odyssey*, vol. 4, 2004, pp. 219–226. [Cited on page 21].

[68] A. Solomonoff, W. M. Campbell, and I. Boardman, "Advances in channel compensation for SVM speaker recognition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2005. [Cited on pages 21, 42].

[69] W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2006. [Cited on pages 21, 42].

[70] H. Aronowitz, "Unsupervised Compensation of Intra-Session Intra-Speaker Variability for Speaker Diarization," in *Proc. Speaker Odyssey*, 2010, p. 25. [Cited on page 21].

[71] P. Kenny and P. Dumouchel, "Disentangling speaker and channel effects in speaker verification," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2004, pp. I–37. [Cited on page 21].

[72] P. Kenny, "Joint factor analysis of speaker and session variability: Theory and algorithms," *CRIM, Montreal,(Report) CRIM-06/08-13*, vol. 14, pp. 28–29, 2005. [Cited on page 21].

[73] N. Dehak, R. Dehak, P. Kenny, N. Brümmer, P. Ouellet, and P. Dumouchel, "Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verification," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2009. [Cited on page 22].

[74] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010. [Cited on page 22].

[75] N. Dehak, "Discriminative and generative approaches for long-and short-term speaker characteristics modeling: application to speaker verification," Ph.D. dissertation, École de technologie supérieure, 2009. [Cited on page 22].

[76] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Proc. INTERSPEECH*, 2011. [Cited on page 23].

[77] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. IEEE International Conference on Computer Vision*, 2007, pp. 1–8. [Cited on page 23].

[78] F. Castaldo, D. Colibro, E. Dalmasso, P. Laface, and C. Vair, "Stream-based speaker segmentation using speaker factors and eigenvoices," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2008, pp. 4133–4136. [Cited on page 23].

[79] P. Kenny, D. Reynolds, and F. Castaldo, "Diarization of telephone conversations using factor analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 6, pp. 1059–1070, 2010. [Cited on page 23].

[80] S. Shum, N. Dehak, E. Chuangsuwanich, D. Reynolds, and J. Glass, "Exploiting intra-conversation variability for speaker diarization," in *Proc. INTERSPEECH*, 2011. [Cited on pages 23, 26].

[81] G. Sell and D. Garcia-Romero, "Speaker diarization with PLDA i-vector scoring and unsupervised calibration," in *Proc. IEEE Spoken Language Technology Workshop (SLT)*, 2014, pp. 413–417. [Cited on pages 23, 136].

[82] G. Sell, A. McCree, and D. Garcia-Romero, "Priors for Speaker Counting and Diarization with AHC," in *Proc. INTERSPEECH*, 2016, pp. 2194–2198. [Cited on page 23].

[83] J. Villalba and E. Lleida, "Unsupervised adaptation of PLDA by using variational Bayes methods," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 744–748. [Cited on page 23].

[84] J. Villalba, A. Ortega, A. Miguel, and E. Lleida, "Variational Bayesian PLDA for speaker diarization in the MGB Challenge," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 667–674. [Cited on page 23].

[85] I. Viñals, A. Ortega, J. A. V. López, A. Miguel, and E. Lleida, "Domain Adaptation of PLDA Models in Broadcast Diarization by Means of Unsupervised Speaker Clustering," in *Proc. INTERSPEECH*, 2017, pp. 2829–2833. [Cited on page 23].

[86] I. Viñals, P. Gimeno, A. Ortega, A. Miguel, and E. Lleida, "Estimation of the Number of Speakers with Variational Bayesian PLDA in the DIHARD Diarization Challenge," in *Proc. INTERSPEECH*, 2018, pp. 2803–2807. [Cited on pages 23, 105].

[87] ——, "In-domain Adaptation Solutions for the RTVE 2018 Diarization Challenge," in *Proc. IberSPEECH*, 2018, pp. 220–223. [Cited on page 23].

[88] G. Le Lan, S. Meignier, D. Charlet, and A. Larcher, "First investigations on self trained speaker diarization," 2016. [Cited on page 23].

[89] G. Le Lan, D. Charlet, A. Larcher, and S. Meignier, "Iterative PLDA adaptation for speaker diarization," 2016. [Cited on page 23].

[90] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1695–1699. [Cited on page 23].

[91] P. Kenny, T. Stafylakis, P. Ouellet, V. Gupta, and M. J. Alam, "Deep Neural Networks for extracting Baum-Welch statistics for Speaker Recognition," in *Proc. Speaker Odyssey*, 2014, pp. 293–298. [Cited on page 24].

[92] D. Snyder, D. Garcia-Romero, and D. Povey, "Time delay deep neural network-based universal background models for speaker recognition," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 92–97. [Cited on page 24].

[93] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *Backpropagation: Theory, Architectures and Applications*, pp. 35–61, 1995. [Cited on page 24].

[94] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proc. INTERSPEECH*, 2015. [Cited on page 24].

[95] G. Sell, D. Garcia-Romero, and A. McCree, "Speaker diarization with i-vectors from DNN senone posteriors," in *Proc. INTERSPEECH*, 2015. [Cited on page 24].

[96] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4879–4883. [Cited on page 24].

[97] G. Le Lan, D. Charlet, A. Larcher, and S. Meignier, "A triplet ranking-based neural network for speaker diarization and linking," in *Proc. INTERSPEECH*, 2017. [Cited on page 24].

[98] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," in *Proc. IEEE Spoken Language Technology Workshop (SLT)*, 2016, pp. 165–170. [Cited on page 24].

[99] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villalba, M. Maciejewski, V. Manohar, N. Dehak, D. Povey, S. Watanabe, *et al.*, "Diarization is hard: Some experiences and lessons learned for the jhu team in the inaugural dihard challenge," in *Proc. INTERSPEECH*, 2018, pp. 2808–2812. [Cited on pages 24, 25, 27, 105, 106, 111, 114].

[100] N. Evans, S. Bozonnet, D. Wang, C. Fredouille, and R. Troncy, "A comparative study of bottom-up and top-down approaches to speaker diarization," *IEEE Transactions on Audio, speech, and language processing*, vol. 20, no. 2, pp. 382–392, 2012. [Cited on pages 25, 146].

[101] S. Meignier, J.-F. Bonastre, and S. Igounet, "E-HMM approach for learning and adapting sound models for speaker indexing," in *Proc. Speaker Odyssey*, 2001. [Cited on page 25].

[102] C. Fredouille and G. Senay, "Technical improvements of the E-HMM based speaker diarization system for meeting records," in *Proc. International Workshop on Machine Learning for Multimodal Interaction*, 2006, pp. 359–370. [Cited on page 25].

[103] C. Fredouille, S. Bozonnet, and N. Evans, "The LIA-EURECOM RT '09 Speaker Diarization System," in *Proc. RT'09, NIST Rich Transcription Workshop*, 2009, pp. 17–23. [Cited on page 25].

[104] C. Wooters, J. Fung, B. Peskin, and X. Anguera, "Towards robust speaker segmentation: The ICSI-SRI fall 2004 diarization system," in *Proc. RT-04F Workshop*, vol. 23, 2004, p. 23. [Cited on page 25].

[105] H. Delgado, "Fast cross-session speaker diarization," Ph.D. dissertation, Autonomous University of Barcelona, 9 2015. [Cited on pages 25, 41, 42, 44, 153].

[106] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," in *Proc. of the Allerton Conference on Communication, Control and Computation*, 2001. [Cited on page 25].

[107] N. Slonim, "The information bottleneck: Theory and applications," Ph.D. dissertation, 2002. [Cited on page 25].

[108] D. Vijayasenan, F. Valente, and H. Bourlard, "An information theoretic approach to speaker diarization of meeting data," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 7, pp. 1382–1393, 2009. [Cited on page 25].

[109] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012. [Cited on page 25].

[110] F. Valente and C. Wellekens, "Variational Bayesian methods for audio indexing," in *Proc. International Workshop on Machine Learning for Multimodal Interaction*, 2005, pp. 307–319. [Cited on page 25].

[111] P. Kenny, "Bayesian analysis of speaker diarization with eigenvoice priors," *CRIM, Montreal, Technical Report*, 2008. [Cited on page 25].

[112] F. Valente, P. Motlicek, and D. Vijayasenan, "Variational Bayesian speaker diarization of meeting recordings," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2010, pp. 4954–4957. [Cited on page 25].

[113] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Sharing clusters among related groups: Hierarchical Dirichlet processes," in *Proc. Advances in neural information processing systems (NeurIPs*, 2005, pp. 1385–1392. [Cited on page 25].

[114] E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky, "The sticky HDP-HMM: Bayesian nonparametric hidden Markov models with persistent states," 2007. [Cited on page 25].

[115] E. B. Fox, E. B. Sudderth, M. I. Jordan, A. S. Willsky, *et al.*, "A sticky HDP-HMM with application to speaker diarization," *The Annals of Applied Statistics*, vol. 5, no. 2A, pp. 1020–1056, 2011. [Cited on page 25].

[116] M. Diez, L. Burget, and P. Matejka, "Speaker diarization based on bayesian hmm with eigenvoice priors," in *Proc. Speaker Odyssey*, vol. 2018, 2018, pp. 147–154. [Cited on pages 25, 26, 27, 89].

[117] M. Diez, F. Landini, L. Burget, J. Rohdin, A. Silnova, K. Zmolıková, O. Novotnỳ, K. Veselỳ, O. Glembek, O. Plchot, *et al.*, "BUT System for DIHARD Speech Diarization Challenge 2018," in *Proc. INTERSPEECH*, 2018, pp. 2798–2802. [Cited on pages 26, 105, 106].

[118] M. Rouvier and S. Meignier, "A global optimization framework for speaker diarization," in *Proc. Speaker Odyssey*, 2012. [Cited on pages 26, 43].

[119] H. Ning, M. Liu, H. Tang, and T. S. Huang, "A spectral clustering approach to speaker diarization," in *Proc. International Conference on Spoken Language Processing (ICSLP)*, 2006. [Cited on pages 26, 89, 95, 96].

[120] S. Shum, N. Dehak, and J. Glass, "On the use of spectral and iterative methods for speaker diarization," in *Proc. INTERSPEECH*, 2012. [Cited on pages 26, 89, 95, 96].

[121] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on information theory*, vol. 21, no. 1, pp. 32–40, 1975. [Cited on page 26].

[122] M. Senoussaoui, P. Kenny, T. Stafylakis, and P. Dumouchel, "A study of the cosine distance-based mean shift for telephone speech diarization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 1, pp. 217–227, 2013. [Cited on page 26].

[123] T. Stafylakis, V. Katsouros, and G. Carayannis, "Speaker clustering via the mean shift algorithm," *Recall*, vol. 2, p. 7, 2010. [Cited on page 26].

[124] T. Stafylakis, V. Katsouros, P. Kenny, and P. Dumouchel, "Mean shift algorithm for exponential families with applications to speaker clustering," in *Proc. Speaker Odyssey*, 2012. [Cited on page 26].

[125] K. Markov and S. Nakamura, "Never-ending learning system for on-line speaker diarization," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2007, pp. 699–704. [Cited on page 26].

[126] H. Aronowitz, Y. A. Solewicz, and O. Toledo-Ronen, "Online two speaker diarization," in *Proc. Speaker Odyssey*, 2012. [Cited on page 26].

[127] W. Zhu and J. Pelecanos, "Online speaker diarization using adapted i-vector transforms," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5045–5049. [Cited on page 26].

[128] G. Soldi, M. Todisco, H. Delgado, C. Beaugeant, and N. W. Evans, "Semi-supervised On-line Speaker Diarization for Meeting Data with Incremental Maximum A-posteriori Adaptation," in *Proc. Speaker Odyssey*, 2016, pp. 377–384. [Cited on page 26].

[129] D. A. Reynolds and P. Torres-Carrasquillo, "Approaches and applications of audio diarization," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 5, 2005, pp. v–953. [Cited on page 26].

[130] G. Sell and D. Garcia-Romero, "Diarization resegmentation in the factor analysis subspace," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4794–4798. [Cited on page 27].

[131] N. Dawalatabad, S. R. Madikeri, C. C. Sekhar, and H. A. Murthy, "Two-Pass IB Based Speaker Diarization System Using Meeting-Specific ANN Based Features," in *Proc. INTERSPEECH*, 2016, pp. 2199–2203. [Cited on page 27].

[132] R. Yin, H. Bredin, and C. Barras, "Neural speech turn segmentation and affinity propagation for speaker diarization," in *Proc. INTERSPEECH*, 2018. [Cited on pages 27, 113].

[133] "NIST Rich Transcription Evaluation," 2009. [Online]. Available: https://www.nist.gov/itl/iad/mig/rich-transcription-evaluation [Cited on page 27].

[134] C. S. Greenberg, A. F. Martin, B. N. Barr, and G. R. Doddington, "Report on performance results in the NIST 2010 speaker recognition evaluation," in *Proc. INTERSPEECH*, 2011. [Cited on pages xiv, 28, 116, 140, 141].

[135] S. Shi, Q. Wang, P. Xu, and X. Chu, "Benchmarking state-of-the-art deep learning software tools," in *Proc. International Conference on Cloud Computing and Big Data (CCBD)*, 2016, pp. 99–104. [Cited on page 35].

[136] J. Rohdin, T. Stafylakis, A. Silnova, H. Zeinali, L. Burget, and O. Plchot, "Speaker verification using end-to-end adversarial language adaptation," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018. [Cited on page 35].

[137] X. Anguera and J.-F. Bonastre, "A novel speaker binary key derived from anchor models," in *Proc. INTERSPEECH*, 2010, pp. 2118–2121. [Cited on pages 35, 36, 39, 42].

[138] T. Merlin, J.-F. Bonastre, and C. Fredouille, "Non directly acoustic process for costless speaker recognition and indexation," in *Proc. Intl. Workshop on Intelligent Communication Technologies and Applications*, vol. 29, 1999. [Cited on page 36].

[139] Y. Mami and D. Charlet, "Speaker identification by location in an optimal space of anchor models," in *Proc. Intl. Conf. on Spoken Language Processing (ICSLP)*, 2002. [Cited on page 36].

[140] X. Anguera and J.-F. Bonastre, "Fast speaker diarization based on binary keys," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 4428–4431. [Cited on pages 36, 38, 42, 43].

[141] J.-F. Bonastre, P.-M. Bousquet, D. Matrouf, and X. Anguera, "Discriminant binary data representation for speaker recognition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 5284–5287. [Cited on pages 38, 42].

[142] G. Hernández-Sierra, J. R. Calvo, and J.-F. Bonastre, "Temporal information in a binary framework for speaker recognition," in *Proc. Iberoamerican Congress on Pattern Recognition*, 2014, pp. 207–213. [Cited on page 42].

[143] P.-M. Bousquet and J.-F. Bonastre, "Typicality extraction in a speaker binary keys model," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 1713–1716. [Cited on page 42].

[144] M. A. Martínez, G. Hernández-Sierra, and J. R. C. de Lara, "Speaker Verification Using Accumulative Vectors with Support Vector Machines," 2013, pp. 350–357. [Cited on page 42].

[145] G. Hernández-Sierra, J.-F. Bonastre, and J. R. C. de Lara, "Speaker recognition using a binary representation and specificities models," in *Proc. Iberoamerican Congress on Pattern Recognition*, 2012, pp. 732–739. [Cited on page 42].

[146] H. Delgado, X. Anguera, C. Fredouille, and J. Serrano, "Improved binary key speaker diarization system," in *Proc. European Signal Processing Conference (EUSIPCO)*, 2015, pp. 2087–2091. [Cited on pages 42, 44, 45].

[147] T. H. Nguyen, E. S. Chng, and H. Li, "T-test distance and clustering criterion for speaker diarization," in *Proc. INTERSPEECH*, 2008. [Cited on page 43].

[148] H. Delgado, X. Anguera, C. Fredouille, and J. Serrano, "Global speaker clustering towards optimal stopping criterion in binary key speaker diarization," in *Proc. IberSPEECH*. Springer, 2014, pp. 59–68. [Cited on page 43].

[149] ——, "Novel Clustering Selection Criterion for Fast Binary Key Speaker Diarization," in *Proc. INTERSPEECH*, pp. 3091–3095. [Cited on pages 43, 46, 60, 64, 71, 92, 93].

[150] H. Delgado, C. Fredouille, and J. Serrano, "Towards a complete binary key system for the speaker diarization task," in *Proc. INTERSPEECH*, 2014, pp. 572–576. [Cited on pages 44, 45].

[151] X. Anguera, E. Movellan, and M. Ferrarons, "Emotions recognition using binary fingerprints," *Proc. IberSPEECH*, 2012. [Cited on page 44].

[152] J. Luque and X. Anguera, "On the modeling of natural vocal emotion expressions through binary key," in *Proc. European Signal Processing Conference (EUSIPCO)*, 2014, pp. 1562–1566. [Cited on page 44].

[153] A. Mtibaa, D. Petrovska-Delacretaz, and A. B. Hamida, "Cancelable speaker verification system based on binary Gaussian mixtures," in *Proc. Advanced Technologies for Signal and Image Processing (ATSIP)*, 2018, pp. 1–6. [Cited on page 44].

[154] A. Nautsch, J. Patino, A. Treiber, T. Stafylakis, P. Mizera, M. Todisco, T. Schneider, and N. Evans, "Privacy-Preserving Speaker Recognition with Cohort Score Normalisation," in *Proc. INTERSPEECH*, 2019. [Cited on page 44].

[155] J. Patino, H. Delgado, N. Evans, and X. Anguera, "EURECOM submission to the Albayzin 2016 Speaker Diarization Evaluation," *Proc. IberSPEECH*, 2016. [Cited on pages 49, 66, 71, 83, 88, 89, 92, 93, 101, 102, 111].

[156] G. Valenti, A. Daniel, N. Evans, N. Semiconductors, and F. Mougins, "End-to-end automatic speaker verification with evolving recurrent neural networks," in *Proc. Speaker Odyssey*, 2018. [Cited on page 50].

[157] J. Youngberg and S. Boll, "Constant-Q signal analysis and synthesis," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1978, pp. 375–378. [Cited on pages 50, 52, 53].

[158] J. C. Brown, "Calculation of a constant Q spectral transform," *The Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991. [Cited on pages 50, 53, 54].

[159] P. Cancela, M. Rocamora, and E. López, "An Efficient Multi-Resolution Spectral Transform for Music Analysis," in *Proc. ISMIR*, 2009, pp. 309–314. [Cited on pages 50, 53, 54].

[160] H. Delgado, M. Todisco, M. Sahidullah, A. K. Sarkar, N. Evans, T. Kinnunen, and Z.-H. Tan, "Further optimisations of constant Q cepstral processing for integrated utterance and text-dependent speaker verification," in *Proc. IEEE Spoken Language Technology Workshop (SLT)*, 2016, pp. 179–185. [Cited on pages 50, 51, 53, 55, 58, 59, 67, 92, 111, 117, 153].

[161] M. Todisco, H. Delgado, and N. Evans, "A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients," in *Proc. Speaker Odyssey*, 2016, pp. 249–252. [Cited on pages 50, 52, 53, 54, 55, 59, 92].

[162] J. Alam and P. Kenny, "Spoofing detection employing infinite impulse response—constant Q transform-based feature representations," in *Proc. European Signal Processing Conference (EUSIPCO)*, 2017, pp. 101–105. [Cited on pages 50, 52].

[163] K. Dressler, "Sinusoidal extraction using an efficient implementation of a multi-resolution FFT," in *Proc. of the Int. Conf. on Digital Audio Effects (DAFx-06)*, 2006, pp. 247–252. [Cited on page 52].

[164] F. C. Diniz, I. Kothe, S. L. Netto, and L. W. Biscainho, "High-selectivity filter banks for spectral analysis of music signals," *EURASIP Journal on Applied Signal Processing*, vol. 2007, no. 1, pp. 164–164, 2007. [Cited on page 52].

[165] B. C. Moore, *An introduction to the psychology of hearing.* Brill, 2012. [Cited on page 53].

[166] M. Todisco, H. Delgado, and N. W. Evans, "Articulation Rate Filtering of CQCC Features for Automatic Speaker Verification," in *Proc. INTERSPEECH*, 2016, pp. 3628–3632. [Cited on pages 53, 54, 55].

[167] J. C. Brown and M. S. Puckette, "An efficient algorithm for the calculation of a constant Q transform," *The Journal of the Acoustical Society of America*, vol. 92, no. 5, pp. 2698–2701, 1992. [Cited on pages 53, 54].

[168] K. Kashima and B. Mont-Reynaud, "The bounded-Q approach to time-varying spectral analysis," *Dept. of Music, Stanford Univ., Tech. Rep. STAN-M-28*, 1985. [Cited on page 53].

[169] H. Schulz, M. R. Costa-Jussa, and J. A. Fonollosa, "TECNOPARLA-Speech technologies for Catalan and its application to Speech-to-speech Translation," *Procesamiento del lenguaje Natural*, vol. 41, pp. 319–320, 2008. [Cited on page 58].

[170] D. Tavarez, X. Sarasola, E. Navas, L. Serrano, A. Alonso, I. Saratxaga, and I. Hernaez, "Aholab Speaker Diarization System for Albayzin 2016 Evaluation Campaign," in *Proc. IberSPEECH*, 2016. [Cited on pages 66, 89].

[171] P. Lopez-Otero, L. Docio-Fernandez, and C. Garcia-Mateo, "GTM-UVigo System for Albayzin 2016 Speaker Diarisation Evaluation," in *Proc. IberSPEECH*, 2016. [Cited on page 66].

[172] P. Ramirez Hereza, J. Franco-Pedroso, and J. Gonzalez-Rodriguez, "ATVS-UAM System Description for the Albayzin 2016 Speaker Diarization Evaluation," in *Proc. IberSPEECH*, 2016. [Cited on page 66].

[173] J. Patino, H. Delgado, and N. Evans, "Speaker Change Detection Using Binary Key Modelling with Contextual Information," in *Proc. International Conference on Statistical Language and Speech Processing (SLSP)*, 2017, pp. 250–261. [Cited on page 69].

[174] C. Barras, X. Zhu, S. Meignier, and J.-L. Gauvain, "Multistage speaker diarization of broadcast news," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1505–1512, 2006. [Cited on page 70].

[175] R. Wang, M. Gu, L. Li, M. Xu, and T. F. Zheng, "Speaker segmentation using deep speaker vectors for fast speaker change scenarios," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5420–5424. [Cited on page 70].

[176] G. Gravier, G. Adda, N. Paulson, M. Carré, A. Giraudel, and O. Galibert, "The ETAPE corpus for the evaluation of speech-based TV content processing in the French language," in *Proc. International Conference on Language Resources and Evaluation (LREC)*, 2012. [Cited on pages 74, 88, 89].

[177] P. Delacourt and C. J. Wellekens, "DISTBIC: A speaker-based segmentation for audio data indexing," *Speech communication*, vol. 32, no. 1, pp. 111–126, 2000. [Cited on page 76].

[178] M. Cettolo and M. Vescovi, "Efficient audio segmentation algorithms based on the BIC," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 6, 2003, pp. VI–537. [Cited on page 76].

[179] S.-S. Cheng, H.-M. Wang, and H.-C. Fu, "BIC-based speaker segmentation using divide-and-conquer strategies with application to speaker diarization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 1, pp. 141–157, 2010. [Cited on page 76].

[180] J. Patino, H. Delgado, and N. Evans, "The EURECOM Submission to the First DIHARD Challenge," in *Proc. INTERSPEECH*, 2018, pp. 2813–2817. [Cited on pages 87, 105, 107].

[181] "The NIST Rich Transcript Evaluation 2009," 2009, https://www.nist.gov/itl/iad/mig/rich-transcription-evaluation. [Cited on page 88].

[182] O. Galibert and J. Kahn, "The first official REPERE evaluation," in *Proc. Workshop on Speech, Language and Audio in Multimedia*, 2013. [Cited on page 88].

[183] I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, *et al.*, "The AMI meeting corpus," in *Proc. International Conference on Methods and Techniques in Behavioral Research*, vol. 88, 2005, p. 100. [Cited on pages 88, 90].

[184] A. Canavan, D. Graff, and G. Zipperlen, "Callhome american english speech," *Linguistic Data Consortium (LDC)*, 1997. [Cited on page 89].

[185] A. Zhang, Q. Wang, Z. Zhu, J. Paisley, and C. Wang, "Fully supervised speaker diarization," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6301–6305. [Cited on page 89].

[186] A. Giraudel, M. Carré, V. Mapelli, J. Kahn, O. Galibert, and L. Quintard, "The repere corpus: a multimodal corpus for person recognition." in *Proc. International Conference on Language Resources and Evaluation (LREC)*, 2012, pp. 1102–1107. [Cited on page 89].

[187] J. S. Garofolo, C. Laprun, M. Michel, V. M. Stanford, and E. Tabassi, "The NIST Meeting Room Pilot Corpus," in *Proc. International Conference on Language Resources and Evaluation (LREC)*, 2004. [Cited on page 90].

[188] J. G. Fiscus, J. Ajot, M. Michel, and J. S. Garofolo, "The rich transcription 2006 spring meeting recognition evaluation," in *International Workshop on Machine Learning for Multimodal Interaction*. Springer, 2006, pp. 309–322. [Cited on page 90].

[189] J. G. Fiscus, J. Ajot, and J. S. Garofolo, "The rich transcription 2007 meeting recognition evaluation," in *Multimodal Technologies for Perception of Humans*. Springer, 2007, pp. 373–389. [Cited on page 90].

[190] C. Fredouille, D. Moraru, S. Meignier, L. Besacier, and J.-F. Bonastre, "The nist 2004 spring rich transcription evaluation: two-axis merging strategy in the context of multiple distance microphone based meeting speaker segmentation," in *Proc. RT 2004 Spring Meeting Recognition Workshop*, 2004. [Cited on page 90].

[191] X. Anguera Miró, "Robust speaker diarization for meetings," Ph.D. dissertation, Universitat Politècnica de Catalunya, 2006. [Cited on pages 90, 106].

[192] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in neural information processing systems (NeurIPs*, 2002, pp. 849–856. [Cited on pages 94, 95, 96].

[193] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010. [Cited on page 95].

[194] D. P. Ellis and J. C. Liu, "Speaker turn segmentation based on between-channel differences," 2004. [Cited on page 95].

[195] D. Verma and M. Meila, "A comparison of spectral clustering algorithms," *University of Washington Tech Rep UWCSE030501*, vol. 1, pp. 1–18, 2003. [Cited on page 96].

[196] D. Marr and E. Hildreth, "Theory of edge detection," vol. 207, no. 1167. The Royal Society London, 1980, pp. 187–217. [Cited on page 97].

[197] M. Basu, "Gaussian-based edge-detection methods-a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 32, no. 3, pp. 252–260, 2002. [Cited on page 97].

[198] R. R. Coifman and S. Lafon, "Diffusion maps," *Applied and computational harmonic analysis*, vol. 21, no. 1, pp. 5–30, 2006. [Cited on page 97].

[199] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Proc. Advances in neural information processing systems (NeurIPs)*, 2005, pp. 1601–1608. [Cited on page 99].

[200] E. Bergelson, "Bergelson Seedlings HomeBank Corpus," 2016. [Cited on page 101].

[201] N. Ryant *et al.*, "DIHARD Corpus," 2018, linguistic Data Consortium. [Cited on page 101].

[202] L. Sun, J. Du, C. Jiang, X. Zhang, S. He, B. Yin, and C.-H. Lee, "Speaker Diarization with Enhancing Speech for the First DIHARD Challenge," *Proc. INTERSPEECH*, pp. 2793–2797, 2018. [Cited on pages 105, 106].

[203] Z. Zajıc, M. Kunešová, J. Zelinka, and M. Hrúz, "ZCU-NTIS Speaker Diarization System for the DIHARD 2018 Challenge," in *Proc. INTERSPEECH*, 2018, pp. 2788–2792. [Cited on page 105].

[204] V. A. Miasato Filho, D. A. Silva, and L. G. Depra Cuozzo, "Joint Discriminative Embedding Learning, Speech Activity and Overlap Detection for the DIHARD Speaker Diarization Challenge," in *Proc. INTERSPEECH 2018*, 2018, pp. 2818–2822. [Cited on page 105].

[205] I. Himawan, M. H. Rahman, S. Sridharan, C. Fookes, and A. Kanagasundaram, "Investigating Deep Neural Networks for Speaker Diarization in the DIHARD Challenge," in *Proc. IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 1029–1035. [Cited on page 105].

[206] S. Meignier, D. Moraru, C. Fredouille, J.-F. Bonastre, and L. Besacier, "Step-by-step and integrated approaches in broadcast news speaker diarization," *Computer Speech & Language*, vol. 20, no. 2-3, pp. 303–330, 2006. [Cited on pages xiv, 110, 111, 114, 115, 116, 122, 124].

[207] S. Bozonnet, N. Evans, X. Anguera, O. Vinyals, G. Friedland, and C. Fredouille, "System output combination for improved speaker diarization," in *Proc. INTERSPEECH*, 2010. [Cited on page 110].

[208] J. Patino, H. Delgado, R. Yin, H. Bredin, C. Barras, and N. Evans, "ODESSA at Albayzin Speaker Diarization Challenge 2018," in *Proc. IberSPEECH, Barcelona, Spain*, 2018. [Cited on page 110].

[209] E. Lleida, A. Ortega, A. Miguel, V. Bazán, C. Pérez, M. Zotano, and A. de Prada, "RTVE2018 Database Description," 2018. [Cited on pages 110, 117].

[210] G. Gelly and J.-L. Gauvain, "Spoken Language Identification using LSTM-based Angular Proximity," in *Proc. INTERSPEECH*, 2017. [Cited on pages 112, 136].

[211] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-Vectors: Robust DNN Embeddings for Speaker Recognition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333. [Cited on page 112].

[212] V. Peddinti, D. Povey, and S. Khudanpur, "A Time Delay Neural Network Architecture for Efficient Modeling of Long Temporal Contexts," in *Proc. INTERSPEECH*, 2015, pp. 3214–3218. [Cited on page 112].

[213] B. J. Frey and D. Dueck, "Clustering by Passing Messages Between Data Points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007. [Cited on page 113].

[214] K. Lee, V. Hautamäki, T. Kinnunen, A. Larcher, C. Zhang, A. Nautsch, T. Stafylakis, G. Liu, M. Rouvier, W. Rao, *et al.*, "The I4U mega fusion and collaboration for NIST speaker recognition evaluation 2016," in *Proc. INTERSPEECH*, 2017. [Cited on page 113].

[215] L. Burget, M. Fapšo, V. Hubeika, O. Glembek, M. Karafiát, M. Kockmann, P. Matějka, P. Schwarz, and J. Černockỳ, "BUT system for NIST 2008 speaker recognition evaluation," in *Proc. INTERSPEECH*, 2009. [Cited on page 113].

[216] K. A. Lee, A. Larcher, H. Thai, B. Ma, and H. Li, "Joint application of speech and speaker recognition for automation and security in smart home," in *Proc. INTERSPEECH*, 2011. [Cited on pages 114, 131].

[217] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: a large-scale speaker identification dataset," in *Proc. INTERSPEECH*, 2017. [Cited on page 116].

[218] P. Kenny, T. Stafylakis, P. Ouellet, M. J. Alam, and P. Dumouchel, "PLDA for speaker verification with utterances of arbitrary duration," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7649–7653. [Cited on page 117].

[219] I. Viñals, P. Gimeno, A. Ortega, A. Miguel, and E. Lleida, "In-domain Adaptation Solutions for the RTVE 2018 Diarization Challenge," in *Proc. IberSPEECH 2018*, 2018, pp. 220–223. [Cited on page 123].

[220] E. L. Campbell, G. Hernandez, and J. R. Calvo de Lara, "CENATAV Voice-Group Systems for Albayzin 2018 Speaker Diarization Evaluation Campaign," in *Proc. IberSPEECH 2018*, 2018, pp. 227–230. [Cited on page 123].

[221] O. Ghahabi and V. Fischer, "EML Submission to Albayzin 2018 Speaker Diarization Challenge," in *Proc. IberSPEECH 2018*, 2018, pp. 216–219. [Cited on page 123].

[222] A. Lozano-Diez, B. Labrador, D. de Benito, P. Ramirez, and D. T. Toledano, "DNN-based Embeddings for Speaker Diarization in the AuDIaS-UAM System for the Albayzin 2018 IberSPEECH-RTVE Evaluation," in *Proc. IberSPEECH 2018*, 2018, pp. 224–226. [Cited on page 123].

[223] Z. Huang, L. P. García-Perera, J. Villalba, D. Povey, and N. Dehak, "JHU Diarization System Description," in *Proc. IberSPEECH 2018*, 2018, pp. 236–239. [Cited on page 123].

[224] D. Castan, M. McLaren, and M. K. Nandwana, "The SRI International STAR-LAB System Description for IberSPEECH-RTVE 2018 Speaker Diarization Challenge," in *Proc. IberSPEECH 2018*, 2018, pp. 208–210. [Cited on page 123].

[225] A. Khosravani, C. Glackin, N. Dugan, G. Chollet, and N. Cannings, "The Intelligent Voice System for the IberSPEECH-RTVE 2018 Speaker Diarization Challenge," in *Proc. IberSPEECH 2018*, 2018, pp. 231–235. [Cited on page 123].

[226] J. Patino, R. Yin, H. Delgado, H. Bredin, A. Komaty, G. Wisniewski, C. Barras, N. Evans, and S. Marcel, "Low-latency speaker spotting with online diarization and detection," in *Proc. Speaker Odyssey*, 2018, pp. 140–146. [Cited on pages 129, 145, 150].

[227] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12–40, 2010. [Cited on pages 130, 131].

[228] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker Verification Using Adapted Gaussian Mixture Models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, 2000. [Cited on pages 130, 136].

[229] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 308–311, 2006. [Cited on page 130].

[230] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint Factor Analysis Versus Eigenchannels in Speaker Recognition," *IEEE Trans. on Audio, Speech, and Language Proc.*, vol. 15, no. 4, pp. 1435–1447, May 2007. [Cited on page 130].

[231] N. Dehak, P. Kenny, R. Dehak, P. Ouellet, and P. Dumouchel, "Front-end factor analysis for speaker verification," *IEEE Trans. on Audio, Speech and Language Proc.*, vol. 19, pp. 788–798, 2011. [Cited on pages 130, 135, 136].

[232] M. A. Przybocki, A. F. Martin, and A. N. Le, "NIST speaker recognition evaluation chronicles - part 2," in *Proc. Odyssey*, June 2006, pp. 1–6. [Cited on page 130].

[233] S. O. Sadjadi, T. Kheyrkhah, A. Tong, C. S. Greenberg, E. S. Reynolds, L. Mason, and J. Hernandez-Cordero, "The 2016 NIST Speaker Recognition Evaluation," 2017, pp. 1353–1357. [Cited on page 130].

[234] A. Sarkar, D. Matrouf, P.-M. Bousquet, and J.-F. Bonastre, "Study of the effect of i-vector modeling on short and mismatch utterance duration for speaker verification," in *Proc. INTERSPEECH*, 2012. [Cited on page 130].

[235] L. Lu and H.-J. Zhang, "Unsupervised speaker segmentation and tracking in real-time audio content analysis," *Multimedia Systems*, vol. 10, no. 4, pp. 332–343, Apr. 2005. [Cited on page 131].

[236] T. Oku, S. Sato, A. Kobayashi, S. Homma, and T. Imai, "Low-latency speaker diarization based on Bayesian information criterion with multiple phoneme classes," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 4189–4192. [Cited on page 131].

[237] W. Zhu and J. Pelecanos, "Online speaker diarization using adapted i-vector transforms," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5045–5049. [Cited on page 131].

[238] D. Dimitriadis and P. Fousek, "Developing On-Line Speaker Diarization System," in *Proc. INTERSPEECH*, 2017, pp. 2739–2743. [Cited on page 131].

[239] G. Soldi, M. Todisco, H. Delgado, C. Beaugeant, and N. Evans, "Semi-supervised On-line Speaker Diarization for Meeting Data with Incremental Maximum A-posteriori Adaptation," in *Proc. Speaker Odyssey*, 2016, pp. 377–384. [Cited on page 131].

[240] M. Zamalloa, L.-J. Rodriguez-Fuentes, G. Bordel, M. Penagarikano, and J.-P. Uribe, "Low-latency online speaker tracking on the AMI corpus of meeting conversations," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2010, pp. 4962–4965. [Cited on page 131].

[241] D. Liu and F. Kubala, "Online speaker clustering," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, 2003, pp. 572–575. [Cited on page 131].

[242] T. Koshinaka, K. Nagatomo, and K. Shinoda, "Online speaker clustering using incremental learning of an ergodic hidden Markov model," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 4093–4096. [Cited on page 131].

[243] A. Larcher, K. A. Lee, B. Ma, and H. Li, "Text-dependent speaker verification: Classifiers, databases and RSR2015," *Speech Communication*, vol. 60, pp. 56–77, 2014. [Cited on page 131].

[244] E. Variani, X. Lei, E. McDermott, I. Lopez-Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 4052–4056. [Cited on page 131].

[245] G. Heigold, I. Lopez-Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5115–5119. [Cited on page 131].

[246] M. McLaren, L. Ferrer, D. Castan, and A. Lawson, "The 2016 Speakers in the Wild Speaker Recognition Evaluation," in *Proc. INTERSPEECH*, 2016, pp. 823–827. [Cited on pages 131, 137].

[247] G. Wisniewski, H. Bredin, G. Gelly, and C. Barras, "Combining Speaker Turn Embedding and Incremental Structure Prediction for Low-Latency Speaker Diarization," in *Proc. INTERSPEECH*, 2017. [Cited on page 136].

[248] J. Carletta, "Unleashing the killer corpus: experiences in creating the multi-everything AMI Meeting Corpus," *Language Resources and Evaluation*, vol. 41, no. 2, pp. 181–190, 2007. [Cited on pages 137, 158].

[249] J. Patino, H. Delgado, and N. Evans, "Enhanced low-latency speaker spotting using selective cluster enrichment," in *Proc. International Conference of the Biometrics Special Interest Group (BIOSIG)*, 2018, pp. 1–5. [Cited on page 145].

[250] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 31–35. [Cited on page 159].

[251] Y. Luo, Z. Chen, and N. Mesgarani, "Speaker-independent speech separation with deep attractor network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 4, pp. 787–796, 2018. [Cited on page 159].

[252] Q. Wang, H. Muckenhirn, K. Wilson, P. Sridhar, Z. Wu, J. Hershey, R. A. Saurous, R. J. Weiss, Y. Jia, and I. L. Moreno, "Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking," *Proc. INTERSPEECH*, 2019. [Cited on page 159].

[253] T. Yoshioka, Z. Chen, C. Liu, X. Xiao, H. Erdogan, and D. Dimitriadis, "Low-latency Speaker-independent Continuous Speech Separation," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6980–6984. [Cited on page 159].

[254] K. Zmolikova, M. Delcroix, K. Kinoshita, T. Ochiai, T. Nakatani, L. Burget, and J. Černockỳ, "SpeakerBeam: Speaker aware neural network for target speaker extraction in speech mixtures," *IEEE Journal of Selected Topics in Signal Processing*, 2019. [Cited on page 159].