Spatial features of reverberant speech: estimation and application to recognition and diarization

by
Pablo Peso Parada

A Thesis submitted in fulfilment of requirements for the degree of
Doctor of Philosophy of Imperial College

Speech and audio processing research
Communications and Signal Processing Group
Department of Electrical and Electronic Engineering
Imperial College London
University of London
2016

Copyright declaration

Declaration of originality

I declare that this thesis and the research to which it refers are the product of my own work under the guidance and supervision of Dr Dushyant Sharma and Dr Toon van Waterschoot and my thesis supervisor Dr Patrick A. Naylor. Any ideas or quotations from the work of others, published or otherwise, are fully acknowledged in accordance with standard referencing practice. The material of this thesis has not been accepted for any degree, and has not been concurrently submitted for the award of any other degree.

Acknowledgment

First of all, I sincerely feel that this thesis should include a list of numerous contributors, including not only those people who have helped me during these last years to achieve this milestone in my career but also relevant people who were there for me much earlier, when I started showing some genuine interest in sound and computers.

I would like to express my sincere gratitude to my supervisors: Patrick A. Naylor, Dushyant Sharma and Toon van Waterschoot. Their immense expertise in the field of the thesis topic has been extremely helpful, but also their motivation and willingness to help at any time and with any given task throughout these years have been really valuable and a key factor in the outcome of this thesis.

I cannot forget the colleagues at Nuance who made the work at the office way easier, the colleagues at Imperial College for the stimulating discussions and the ones in KU Leuven who helped me during my secondment and from whom I learnt a lot. Finally, I am grateful to all DREAMS fellows for every suggestion and discussion we shared and of course for the fun they brought during the last three years. I am proud of the team we have become and I think "DREAMS" should never stop.

Of course, I want to also acknowledge my family for being always supportive in my life and last but not the least (at all!) I would like to give a big thank you to Bego for her love, encouragement and editing assistance devoted to this thesis.

For all of you, thank you and GRACIAS!

Abstract

Distant talking scenarios, such as hands-free calling or teleconference meetings, are essential for natural and comfortable human-machine interaction and they are being increasingly used in multiple contexts. The acquired speech signal in such scenarios is reverberant and affected by additive noise. This signal distortion degrades the performance of speech recognition and diarization systems creating troublesome human-machine interactions.

This thesis proposes a method to non-intrusively estimate room acoustic parameters, paying special attention to a room acoustic parameter highly correlated with speech recognition degradation: *clarity index*. In addition, a method to provide information regarding the estimation accuracy is proposed.

An analysis of the phoneme recognition performance for multiple reverberant environments is presented, from which a confusability metric for each phoneme is derived. This confusability metric is then employed to improve reverberant speech recognition performance. Additionally, room acoustic parameters can as well be used in speech recognition to provide robustness against reverberation. A method to exploit *clarity index* estimates in order to perform reverberant speech recognition is introduced.

Finally, room acoustic parameters can also be used to diarize reverberant speech. A room acoustic parameter is proposed to be used as an additional source of information for single-channel diarization purposes in reverberant environments. In multi-channel environments, the time delay of arrival is a feature commonly used to diarize the input speech, however the computation of this feature is affected by reverberation. A method is presented to model the time delay of arrival in a robust manner so that speaker diarization is more accurately performed.

Contents

List of Figures

List of Tables

List of Abbreviations

ACE Acoustic Characterisation of Environments. 30, 31, 34, 75–79, 81, 86

aRMSD average Root Mean Square Deviation. 101, 102, 104

ASR Automatic Speech Recognition. 24–26, 28–31, 33–36, 39–41, 54, 64, 87–96, 98, 104–111, 114, 117, 119, 120, 126, 129, 167–169

BIC Bayesian Information Criterion. 132

BLSTM Bidirectional Long-Short Term Memory. 29, 48, 53, 56, 57, 59, 61, 63, 65, 70, 75, 77, 78, 83, 85, 86

C_{50} Clarity Index. 25, 26, 28–33, 37–41, 45–51, 54–56, 59, 61–66, 68–71, 74, 75, 77, 87, 94, 95, 98–102, 104, 105, 107, 110–114, 116–120, 124–127, 157, 167–169

CART Classification And Regression Trees. 46, 47, 53–57

CD-KALDI Context-Dependent GMM-HMM phone recognizer based on Kaldi toolkit. 94, 95, 99, 106

CI-HTK Context-Independent GMM-HMM phone recognizer based on HTK. 93, 99

CI-KALDI Context-Independent GMM-HMM phone recognizer based on Kaldi toolkit. 94, 99

ClnDev Non-reverberant development set. 94, 95, 97

ClnEval Non-reverberant evaluation set. 94

CMLLR Constrained Maximum Likelihood Linear Regression. 126, 127, 168

D_{50} Definition. 31–33, 37, 39

DBN Deep Belief Network. 47, 48, 51–53, 59

DER Diarization Error Rate. 138, 153

DNN Deep Neural Network. 89, 92, 169

DOA Direction of arrival. 132

DRR Direct-to-Reverberation-Ratio. 26, 29–32, 34, 35, 38, 41, 75, 76, 79, 81–83, 85, 86, 92, 128, 133–135, 137–142, 157, 167–169

DSS Deep Scatter Spectrum. 42, 44, 55–57, 59, 77

EM Expectation-Maximization. 29, 146, 169

GA Genetic Algorithm. 51, 53, 152

GCC-PHAT Generalized Cross Correlation with Phase Transform. 26, 128, 132, 144

GMM Gaussian Mixture Models. 35, 37, 89, 90, 92, 134–136, 146

HLDA Heteroscedastic Linear Discriminant Analysis. 110, 118, 124–127, 168

HMM Hidden Markov Models. 35, 37, 89, 90, 106, 116, 120, 154–156, 160, 162, 164

HTK Hidden Markov Model Toolkit. 104, 106, 108, 120

IB Information Bottleneck. 133, 134, 136, 138, 139

IQR Interquartile range. 79, 80, 85, 86, 148

iSNR importance weighted Signal to Noise Ratio. 44

LDA Linear Discriminant Analysis. 115, 131

LPC Linear Prediction Coefficients. 44, 77

LR Linear Regression. 53

LSF Line Spectrum Frequency. 44, 77

LSTM Long-Short Term Memory. 48

LTASS Long Term Average Speech Spectrum. 44, 45

LVCSR Large Vocabulary Continuous Speech Recognition. 88, 90

MAP Maximum A Posteriori. 153, 154

MD Modulation Domain. 42, 46, 53, 55–57, 59, 70

MFCC Mel-Frequency Cepstral Coefficients. 26, 44, 45, 55, 114, 120, 124, 128, 130–134, 138, 139, 141, 142, 168, 169

MLE Maximum Likelihood Estimate. 33, 34, 146

NIRA Non-Intrusive Room Acoustic estimation. 28, 29, 41, 46, 61, 62, 64, 70, 74, 75, 77–82, 85, 86, 128, 135, 136, 140, 167–169

NIRA-BLSTM Non-Intrusive Room Acoustic estimation using bidirectional long-short term memory. 111, 113, 119, 120, 124–127

NIRA-CART Non-Intrusive Room Acoustic estimation using Classification And Regression Trees. 111, 113, 119, 120, 124–127

NMPIW Normalized Mean Prediction Interval Width. 69–71

OG Optimal Geometry baseline. 143, 160, 165

PER Phoneme Error Rate. 35–39, 94, 108, 110

PESQ Perceptual Evaluation of Speech Quality. 35, 37–39

PI Prediction Interval. 68, 69

PICP Prediction Interval Coverage Probability. 69–71

PLD Power Spectrum of long term Deviation. 45

PLP Perceptual Linear Predictive. 130, 131

PSD Power Spectral Density. 34, 54

rΔ relative difference of the argument. 97

rΔPER relative Phoneme Error Rate degradation. 94

RevDev Reverberant development set. 94, 95, 97, 100, 104

RevEval Reverberant evaluation set. 94, 100, 102, 104

RIR Room Impulse Response. 22, 23, 26, 28, 30–33, 36, 37, 50, 51, 56, 62, 65, 66, 71, 74–76, 78, 82, 86, 91–94, 104, 107, 111, 112, 116, 117, 133, 135–137, 142, 157

RMSD Root Mean Square Deviation. 49, 56, 57, 59, 61–63, 75, 76, 79, 80, 83, 85, 86, 113, 127, 140

RNN Recurrent Neural Network. 29, 48, 90

RReliefF Regressional ReliefF method. 54–56

SNR Signal-to-Noise Ratio. 50, 51, 62, 76, 136, 137

SSE Sum of Squared Errors. 83, 85

SSPE Sum of Squared Percentage Errors. 85

STFT Short Time Fourier Transform. 42, 85, 134

SVR Support Vector Regressor. 34, 78, 79, 85

T_{60} Reverberation time. 26, 29–35, 39, 41, 75, 76, 80, 82, 83, 85, 86, 91, 92, 111, 112, 137, 140, 142, 157, 167, 168

TDOA Time Delay of Arrival. 25, 26, 29, 128, 130–132, 142–146, 148, 150, 151, 153, 155, 160, 163, 165, 168, 169

Ts Centre time. 31, 32, 35, 37, 39

VAD Voice Activity Detector. 45, 135, 139, 160

WER Word Error Rate. 110, 119, 120, 124–127, 168

WERR Word Error Rate Reduction. 126, 127

WFST Weighted Finite-State Transducers. 106

ZCR Zero-Crossing Rate. 44

List of Symbols

A	Random variable. 36, 49
B	Random variable. 36, 49
C	Constant constraint on the mean. 147
E_d	Energy of the direct path in the room impulse response. 32
F	Feature stream. 135
$I(A;B)$	Mutual information of A and B. 36, 49
J	Number of TDOA streams. 144, 145, 151–154
M	Effective length of h(m). 23, 51, 133
N_w	Number of samples in the rectangular window. 75, 135, 149
N_{RIR}	Number of room impulse responses. 116, 117
N_{TDOA}	Number of TDOA samples. 145, 146, 149, 153
$N_{T_kR_k}$	Number of times the phoneme label T_k is classified as R_k. 99
N_ζ	Number of samples in the room impulse response from the beginning to ζ ms after the reception of the direct path. 32
N_{cnd}	Number of different reverberant conditions. 102
N_{cor}	Number of correct labels. 97, 98, 108
N_{del}	Number of deletions. 35, 97, 98, 108, 119

N_{feat}	Number of features in the feature vector. 47
N_{fp}	Number of free parameters to be estimated. 153
N_{frm}	Number of frames. 63, 67
N_{ins}	Number of insertions. 35, 97, 98, 108, 119
N_{mic}	Number of microphones. 144, 145
N_o	Number of overlapped frames. 150, 151
N_{phn}	Number of phonemes. 35, 97–99
N_{sam}	Number samples. 137
N_{sinc}	Number of sinc sidelobes. 32
N_{spk}	Number of speakers. 133, 146–148, 151, 152, 154–156
N_{sub}	Number of substitutions. 35, 97, 98, 108, 119
N_{utt}	Number of utterances. 36, 47, 49, 76, 113, 116
N_{wrd}	Number of words. 119
R_k	Recognized phoneme of class k. 98, 99, 104
T_k	True phoneme of class k. 96, 98, 99
$Y_1(f)$	Fourier transform of an input signal. 144
$Y_2(f)$	Fourier transform of an input signal. 144
$\Omega_{low,u}$	Lower bound of the prediction interval for the uth utterance. 68, 69
$\Omega_{up,u}$	Upper bound of the prediction interval for the uth utterance. 68, 69
Φ	Input per-utterance feature vector. 45–47, 53, 55, 56
Ξ_m	Uncertainty estimating C_{50} due to model limitations. 67
Ξ_t	Total uncertainty estimating C_{50}. 67, 69

Ξ_v	Uncertainty estimating C_{50} due to data limitations. 67
α_u	Phoneme error rate score of the uth utterance. 36, 49
$\bar{\boldsymbol{o}}_l$	Transformed feature vector. 115
β_t	Trade-off parameter. 134
β_u	Measure of reverberation value of the uth utterance. 36, 49
$\boldsymbol{C}$	Vector with the constant constraints on the mean. 146, 148
$\boldsymbol{\Upsilon}$	Vector that defines the standard deviation vector given the constraints. 149
$\boldsymbol{\beta}$	Vector that defines the mean vector given the constraints. 148
$\boldsymbol{\lambda}$	A priori vector. 146
$\boldsymbol{\mu}$	Mean vector. 146
$\boldsymbol{\pi}$	Initial states probabilities. 155
$\boldsymbol{\sigma}$	Standard deviation vector. 146
$\boldsymbol{\tau}$	Vector of TDOA estimates. 146–148, 152
$\boldsymbol{\tau}_o$	Vector of overlapped TDOA estimates between two channels. 150
$\boldsymbol{\theta}$	Model parameters. 135, 146, 149, 151, 152, 169
$\boldsymbol{\theta}_i$	Model parameters of speaker i. 145, 146, 156
$\boldsymbol{d}$	Speaker decision vector. 151, 152
$\boldsymbol{H}$	Matrix with on room impulse response per row. 117
$\boldsymbol{\mathcal{W}}$	Word sequence. 88, 89, 105, 106
$\boldsymbol{o}_l$	Input feature vector. 90, 115

η	Sinc offset considered to find the maximum energy of the direct path. 32
γ	Skewness. 45
κ	Kurtosis. 45
λ_i	A priori of speaker i. 146
$\mathbf{O}$	Input feature vector sequence. 88, 89, 105, 106
$\mathbf{S}_B$	Between-class scatter matrix. 115
$\mathbf{S}_W$	Within-class scatter matrix. 115
$\mathbf{W}$	Matrix of dimension $q_r \times q_c$. 115
$\breve{\mathbf{W}}$	Matrix of dimension $q_r \times q_c$. 115
$\mathcal{A}$	Acoustic model. 116, 117
$\mathcal{B}$	Set of relevance variable. 133, 134, 136
$\mathcal{CF}(T_k,R_k,\mathrm{C}_{50})$	Confusability Factor. 98, 100–102, 105
$\mathcal{C}$	Set of clusters. 133, 134
$\mathcal{G}$	Matrix with the standard deviation constraints. 148
$\mathcal{J}$	Stream feature index. 135
$\mathcal{K}$	Tuning parameter that defines the width of the intervals. 68–71, 74
$\mathcal{M}$	Matrix with the mean constraints. 146
$\mathcal{S}$	HMM state sequence. 90
$\mathcal{S}_l$	HMM state for the lth frame. 90
$\mathcal{V}$	Frame size. 77
$\mathcal{W}_\mathcal{J}$	Weight for the $\mathcal{J}$th feature stream. 136
$\mathcal{Y}$	Uniform linear segmentation of the recorded signal y_n. 133, 134, 136
$\breve{\mathcal{A}}$	Optimal acoustic model for a given reverberant environment. 115

$\mathfrak{K}$	Gaussian kernel transformation. 147
CM_l	Confidence measure for the lth frame. 155
CM_u	Confidence measure for the uth utterance. 68, 70, 72
$\mathrm{C}_{50,u}$	Ground truth C_{50} for the uth utterance. 47, 49, 65–67, 69, 72, 113
$\mathrm{C}_{50,u}(y_u)$	C_{50} observable in the reverberant signal y_u. 65–67
DRR_u	Ground truth DRR for the uth utterance. 76
R_{eval}	Total C_{50} range observed in the evaluation dataset. 69
R_{tr}	Total C_{50} range observed in the training dataset. 68, 69
$\mathrm{T}_{60,\mathrm{u}}$	Ground truth T_{60} for the uth utterance. 76
$\widehat{\mathrm{DRR}_\mathrm{u}}$	Estimated DRR for the uth utterance. 76
$\widehat{\mathrm{T}_{60,\mathrm{u}}}$	Estimated T_{60} for the uth utterance. 76
μ	Mean. 45, 49, 57, 59, 146
μ_i	Mean of speaker i. 146
$\nu(n)$	Additive noise. 133
$\nu_p(n)$	Additive noise present at pth microphone. 142
$\overline{\alpha}$	Average of the phoneme error rate scores. 36
$\overline{\beta}$	Average of a particular measure of reverberation. 36
ϕ	Input per-frame feature vector. 45, 46, 53, 55, 56, 70
π	Initial states probability. 90
ψ	C_{50} threshold for the acoustic model switching method. 116, 117

ρ	Correlation coefficient. 36, 49
σ	Standard deviation. 45, 49, 57, 59, 147, 148
σ_i	Standard deviation of speaker i. 146
τ	TDOA estimate. 146–148
τ_l	TDOA estimate for the frame l. 144, 145, 147, 156
ϱ	Regularization parameter. 47
ϑ	Linear regression coefficients. 47
$\widehat{\Xi_{t,u}}$	Estimation of the total uncertainty Ξ_t for the uth utterance. 67
$\widehat{\mathcal{CF}}(T_k,R_k,\mathrm{C}_{50})$	Estimated confusability factor using a polynomial function. 100, 102
$\widehat{\mathrm{C}_{50,l,u}}(y_u)$	C_{50} estimated at frame l from the reverberant signal y_u. 65, 67, 68, 71
$\widehat{\mathrm{C}_{50,u}}$	Estimated C_{50} for the uth utterance. 47, 49, 113
$\widehat{\mathrm{C}_{50,u}}(y_u)$	C_{50} estimated per utterance from the reverberant signal y_u. 65, 67, 69, 70, 72
ζ	Time index. 32, 35, 37–39
a	Acoustic model index. 117
a_{qr}	Transition probability from state q to state r. 155
b_q	Observation probability of state q. 155
c	Reverberant condition index. 104
fs	Sampling frequency. 32
$h(m)$	Room impulse response. 23, 32, 75
$h_i(m)$	Room impulse response corresponding to ith speaker. 133, 135
h_u	uth room impulse response. 117

$h_{i,p}(m)$	Room impulse response between ith speaker and pth microphone. 142
i	Speaker index. 133, 135, 142, 145, 154–156
j	TDOA channel index. 151–154, 156
k	Phoneme index. 95, 96, 98, 101, 104
l	Frame index. 63, 65, 90, 115, 134, 144, 145, 155
n	Discrete time index. 23, 133, 142
n_d	Direct path sample. 32, 75, 135
p	Microphone index. 142
$s(n)$	Source signal. 23
s_u	uth source signal. 117
u	Utterance index. 47, 49, 67, 68, 76, 117
v	Window analysis index. 150–152
$x_{i,p}(n)$	Reverberant signal present in the pth microphone created by ith speaker. 142
$y(n)$	Reverberant signal. 23, 133, 134
$y_p(n)$	Reverberant signal capture with pth microphone. 142
y_u	uth reverberant signal. 65–67, 117
$N_{\mathcal{A}}$	Number of available acoustic models. 116, 118, 126
$\mathcal{L}$	Cost function. 47
C_ζ	Ratio of the energy in the ζ first milliseconds after the direct path over the remainder in the room impulse response. 35, 37–39, 54

D_ζ	Ratio of the energy in the ζ first milliseconds after the direct path over the all energy in the room impulse response. 35, 37–39

Chapter 1

Introduction

Speech is an acoustic signal primary created by the human vocal chords which propagates through air. It constitutes a powerful communication mechanism, if not the main one, used by humans in everyday interaction. Furthermore, speech is also becoming in recent years an important form of communication with machines such as robots or smart devices. The medium in which the speech wave is propagated plays a key role in the quality of the received signal, thus severely compromising its intelligibility. This degradation is mainly due to different types of noise present in the medium, as for example noise created by air condition systems in case the propagation medium corresponds to air.

Human-machine interactions are being increasingly used in distant-talk scenarios which provide natural and flexible communications. In such scenarios the speaker interacts with a device that is far from the speaker. In enclosed spaces, this propagation may follow multiple paths from the speaker position to the receiver due to reflections from surfaces in the room, in addition to direct path propagation. These reflections create a convolutive distortion at the receiver known as *reverberation* (Fig. 1.1). The term convolutive refers to the fact that this distortion has a linear dependence with the signal emitted in previous instants. Therefore, the sound in the room persists for a period of time after the sound source is dropped.

The reverberation level present in the received signal is determined by the Room Impulse Response (RIR), which depends on the acoustic characteristics of the given en-

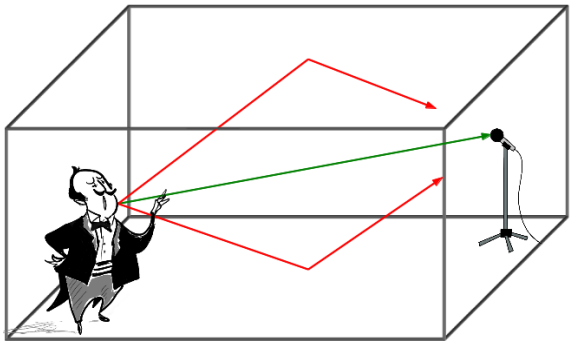

Figure 1.1: Simplified multipath sound propagation example. Green line represents the direct path and red lines represent the reflections.

closure as well as the position of the source and receiver. The reverberant sound $y(n)$ measured at a receiver in the room can be modelled as the convolution of the RIR $h(m)$ and the source signal in the room $s(n)$ so that for each time index n

$$y(n) = \sum_{m=0}^{M-1} h(m)s(n-m) \tag{1.1}$$

where M is the effective length of $h(m)$. The effective length represents the number of samples of the finite RIR considered in the convolution of the input signal $s(n)$. In (1.1) the RIR $h(m)$ is assumed time-invariant, i.e. the position of the source and receiver and the room properties such as the air temperature and density are fixed while $s(n)$ is received. Additionally, reverberation is considered as a linear system in (1.1) although non-linearities may appear in high frequencies or high sound pressure levels.

Typical RIRs can be divided into three different parts as shown in Figure 1.2: the direct path; the early reflections include high magnitude impulses and correspond to approximately the first 50 ms after the direct path depending on the RIR; and late reverberation corresponds to reflections that are delayed approximately more than 50 ms after the direct path and contains lower magnitude impulses and higher temporal density of impulses compared to the early reflection impulses [1]. Early reflections cause spectral coloration of the signal, whereas late reverberation causes temporal smearing and characteristic ringing echoes of the signal [2].

Reverberation can degrade human speech intelligibility [4] or speech quality [5] as

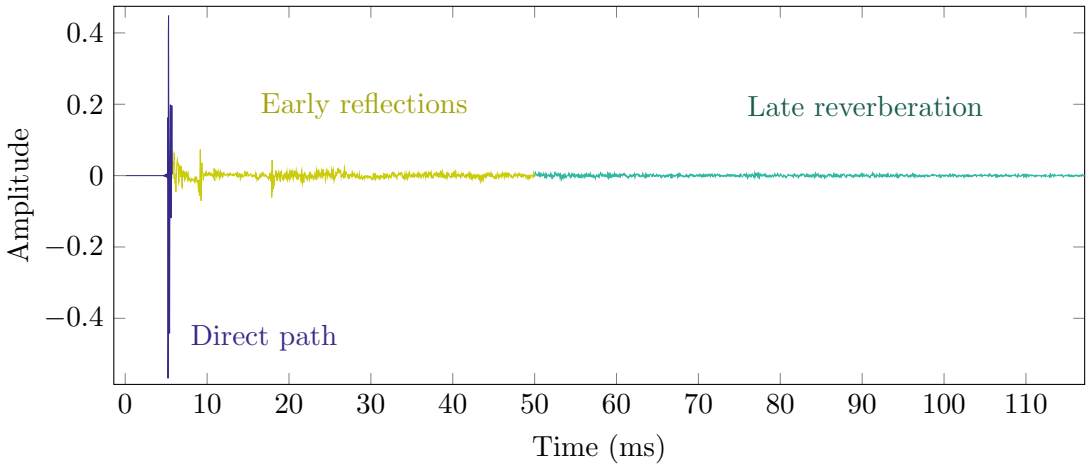

Figure 1.2: Room impulse response measurement from MARDY database [3]. The distance between speaker and microphone is 1 m.

well as potentially reduce Automatic Speech Recognition (ASR) [6] or speech diarization [7] performance. The significance of this degradation highly depends on the magnitude and the delay of the reflections with respect to the direct path. As a result, the functionality of distant-talk applications such as hands-free communications is compromised in reverberant environments.

1.1 Research challenges

This thesis aims to design methods that contribute towards improving the robustness of speech recognition and diarization in reverberant environments. The fundamental challenges to be taken into consideration are:

- Multiple measures of reverberation have been proposed in the literature, nevertheless it is important to know which of these measures is more correlated with the ASR performance. Thus, **finding the measure of reverberation most correlated with ASR** can help not only to predict the ASR performance but to improve the performance of reverberant speech recognition.

- Measures of reverberation need additional information, such as the room character-

istics, in order to be computed. In most scenarios this information is unavailable, therefore **a method to non-intrusively estimate measures of reverberation from single-channel recordings needs to be developed**.

- This estimation needs to be sufficiently accurate and robust to multiple noisy conditions to be potentially integrated in different applications:
  - In the ASR context, reverberant speech recognition can leverage reverberation measure estimates to improve its accuracy. Consequently, **methods that exploit this information to improve ASR performance need to be proposed**.
  - In the diarization context, **measures of reverberation can be used to perform diarization of reverberant multi-party meeting recordings**. In order to successfully perform this task, novel approaches need to be designed.
- A spatial feature commonly used in multi-channel diarization systems is the Time Delay of Arrival (TDOA), however this feature may be highly noisy in reverberant environments due to the multipath sound propagation. Therefrom, **a robust method to process the TDOAs in order to perform diarization in reverberant environments is required**.

This work fits into a larger context of (de)reverberation research network named Dereverberation and Reverberation of Audio, Music, and Speech (DREAMS) network which includes multiple research topics such as dereverberation method tailored to hearing aids, echo cancellation, efficient parametric room acoustic modelling, speech intelligibility analysis for noisy reverberant environments or blind system identification amongst others.

1.2 Structure of the thesis

The remainder of the thesis is organized as follows:

- In Chapter 2, evidence using different set-ups that Clarity Index (C_{50}) is the most correlated parameter to ASR performance among different room acoustic parameters

is provided. Motivated by this finding, a framework to non-intrusively estimate C_{50} is proposed and evaluated using an extensive database including measured RIRs and different noise conditions. Additionally, a confidence measure approach for the C_{50} estimates is investigated. Finally, this framework to predict C_{50} is adapted to estimate Reverberation time (T_{60}) and Direct-to-Reverberation-Ratio (DRR) and evaluated within the ACE Challenge.

- The impact of reverberation on phoneme recognition for multiple reverberant environments is analysed in Chapter 3. From this analysis, a metric to estimate the confusability of each phoneme depending on the reverberation level is derived. This metric is then employed to improve ASR performance. In addition an acoustic model switching method based on C_{50} estimation is introduced to recognize reverberant speech.

- In Chapter 4, two methods to perform diarization from the input speech signal are presented: a single-channel approach based on Mel-Frequency Cepstral Coefficients (MFCC) features and DRR estimation; and a multi-channel approach based on statistically modelling in a robust manner the TDOA estimates obtained with the Generalized Cross Correlation with Phase Transform (GCC-PHAT) algorithm on pairs of microphones.

- The thesis conclusions and suggestions for future work are presented in Chapter 5.

1.3 Thesis outcomes

The following lists show the publications related to the research presented in this thesis:

1.3.1 Journal publications

[J1] P. Peso Parada, D. Sharma, J. Lainez, D. Barreda, T. van Waterschoot, and P. A. Naylor, "A single-channel non-intrusive C50 estimator correlated with speech recognition performance," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 4, pp. 719–732, April 2016

[J2] P. Peso Parada, D. Sharma, P. A. Naylor, and T. van Waterschoot, "Reverberant speech recognition exploiting clarity index estimation," *EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, 2015

[J3] P. Peso Parada, D. Sharma, T. van Waterschoot, and P. A. Naylor, "Confidence measures for non-intrusive estimation of speech clarity index," *The Journal of the Audio Engineering Society*, 2016, Submitted

[J4] A. H. Moore, P. Peso Parada, and P. A. Naylor, "Speech enhancement evaluation using speech recognition," *Computer Speech and Language*, 2016, Submitted

1.3.2 Conference & Workshops publications

[C1] P. Peso Parada, D. Sharma, and P. A. Naylor, "Non-intrusive estimation of the level of reverberation in speech," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 4718–4722

[C2] P. Peso Parada, D. Sharma, P. A. Naylor, and T. van Waterschoot, "Single-channel reverberant speech recognition using C50 estimation," in *Proc. REVERB Challenge*, Florence, Italy, May 2014

[C3] P. Peso Parada, D. Sharma, J. Lainez, D. Barreda, P. A. Naylor, and T. van Waterschoot, "A quantitative comparison of blind C50 estimators," in *Proc. Intl. Workshop Acoust. Signal Enhancement (IWAENC)*, Juan les Pins, France, September 2014, pp. 298–302

[C4] P. Peso Parada, D. Sharma, P. A. Naylor, and T. van Waterschoot, "Reverberant speech recognition: A phoneme analysis," in *Signal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on.* IEEE, December 2014, pp. 567–571

[C5] M. Hu, P. Peso Parada, D. Sharma, S. Doclo, T. van Waterschoot, M. Brookes, and P. A. Naylor, "Single-channel speaker diarization based on spatial features," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics.* IEEE, October 2015, pp. 1–5

[C6] P. Peso Parada, D. Sharma, T. van Waterschoot, and P. A. Naylor, "Evaluating the non-intrusive room acoustics algorithm with the ACE challenge," in *ACE Challenge Workshop, a satellite event of IEEE-WASPAA 2015*, October 2015

[C7] P. Peso Parada, D. Sharma, P. A. Naylor, and T. van Waterschoot, "Analysis of prediction intervals for non-intrusive estimation of speech clarity index," in *Audio Engineering Society Conference: 60th International Conference: DREAMS (Dereverberation and Reverberation of Audio, Music, and Speech)*, February 2016

1.3.3 Patents

[P1] D. Sharma, P. A. Naylor, and P. Peso Parada, "Method for non-intrusive acoustic parameter estimation," Patent. U.S. 20150073780. Mar. 2015

[P2] P. Peso, D. Sharma, P. A. Naylor, and U. Jost, "Microphone selection and multi-talker segmentation with application to ambient automatic speech recognition (ASR)," U.S. Provisional Pat. Ser. No. 62/394,286, filled. September. 2016

The contributions contained in [J4] are not described in this thesis.

1.3.4 Statement of originality

The following aspects of the thesis are, as far as the author is aware, original contributions:

- Analysis of ASR performance dependence with different measures of reverberation computed from mel-frequency bands of the RIRs (Section 2.2.4.2, published in [J1]).
  - The analysis is performed using correlation and mutual information metrics.
- Development of a data-driven framework (Non-Intrusive Room Acoustic estimation (NIRA)) to non-intrusively estimate C_{50} from single-channel noisy reverberant recordings (Section 2.4, published in [J1][C1][C2]).
  - This framework includes novel features based on modulation domain and deep scatter spectrum (Section 2.3.1).

- Development of prediction intervals and confidence measures for NIRA framework (Section 2.5, published in [J3][C7])
  - Prediction intervals and confidence measures are computed from the per frame NIRA estimates.
- Extension of NIRA to estimate DRR and T_{60} (Section 2.6, published in [C6]).
  - Estimation of DRR and T_{60} using Bidirectional Long-Short Term Memory (BLSTM)-Recurrent Neural Network (RNN).
- Reverberant speech recognition based on switching acoustic models using only C_{50} estimates and employing these estimates as an additional ASR input feature (Section 3.4, published [C2][J2]).
  - The C_{50} estimates are computed using NIRA.
- Analysis of the effect from reverberation of individual phonemes and the corresponding impact on ASR (Section 3.2, published in [C4]).
  - Proposed the confusability factor to measure the confusion of the phonemes depending on the level of the reverberation (Section 3.2.3).
- Reverberant speech recognition using the confusability factor (Section 3.3).
  - Scaling of ASR observation probabilities according to the confusability factor.
- Exploiting DRR estimates with the aim of performing single-channel diarization in reverberant environments.
  - The DRR measure is computed using NIRA. This is a joint contribution.
- TDOA modelling for multi-channel diarization tasks employing the Expectation-Maximization (EM) approach with constraints on the mean and variance of the Gaussian models (Section 4.3.2, published [P2]).
  - For each TDOA stream, a Gaussian model is computed for each speaker in addition to a background model.

Chapter 2

Non-intrusive room acoustic parameter estimation

In this chapter, the room acoustic parameters and different methods proposed in the literature to estimate these parameters are first introduced in Section 2.1. In Section 2.2 evidence using different set-ups, ASR engines and measured RIRs that C_{50} is the most correlated parameter to ASR performance is provided. Then in Section 2.3, a framework is described to non-intrusively estimate C_{50} and, in Section 2.4, the C_{50} estimator is evaluated using an extensive database including measured RIRs and different noise conditions. Additionally, in Section 2.5 an approach to estimate the confidence measure of the C_{50} estimates is investigated. Finally, this framework to predict C_{50} is extended to estimate T_{60} and DRR and then evaluated within the Acoustic Characterisation of Environments (ACE) Challenge [21].

The research presented in this chapter relates in part to the following publications [12, 14, 8, 17, 18, 10]

2.1 Introduction

Room acoustic parameters measure different aspects of the reverberation effect present in an enclosed space. Such measurements are employed more often in the last few years in

multiple scenarios where reverberation is involved, e.g. intelligibility estimation of reverberant speech, dereverberation algorithms or reverberant speech recognition. Motivated by these applications several methods have been proposed in the literature to estimate different room acoustic parameters. Many of them have been recently evaluated in the ACE Challenge[1] [21].

2.1.1 Technical background and literature review

In enclosed acoustic spaces such as rooms, sound emitted from a source propagates directly through the air towards the listening position and also reflects off the walls and different objects in the room creating the effect known as reverberation. The energy associated with the reflected waves determines the reverberation level in the room and is often quantified relative to the energy at the receiver due to direct path propagation. Reverberation is known to degrade ASR performance [6] and it is therefore highly valuable to be able to quantify the relation between the reverberation level and ASR performance.

Several room acoustic parameters derived from the RIR have been proposed in the literature [22] [1] in order to measure the level of reverberation. The reverberation time T_{60} is a widely used metric that characterizes the room acoustics properties and it is defined as the time needed for the sound pressure level in the room to drop 60 dB after the acoustic excitation ceases [22]. Assuming an exponential energy decay of the RIR, T_{60} may be computed by fitting a straight line to the smoothed logarithmic energy decay of the RIR. However, the presence of noise floor at the end of the measurement or non-linear logarithmic energy decays with two-stage decay due to the early and late reverberation causes inaccurate T_{60} calculation. In this work it is computed following [23] based on a non-linear optimization of the model with a exponential decay plus a stationary noise floor. Alternative parameters, such as the DRR [22], the Definition (D_{50}) [22], the C_{50} [22] or Centre time (Ts) [1], provide further measures describing the reverberation level in a

[1] ace-challenge.org

signal. The parameter DRR is calculated as [24]

$$\mathrm{DRR} = 10\log_{10}\left(\frac{E_d}{\left(\sum_{m=0}^{M-1} h^2(m)\right) - E_d}\right)\mathrm{dB}, \tag{2.1}$$

where E_d is the direct path energy. Since the direct path may be located between two samples and therefore its energy spread over the adjacent samples, the direct path energy is computed by convolving the squared sinc function with the squared RIR around the direct path sample n_d, given by

$$E_d = \max_{\eta} \sum_{m=-N_{sinc}}^{N_{sinc}} \left(\mathrm{sinc}(m+\eta)h(m+n_d)\right)^2, \tag{2.2}$$

where $N_{sinc} = 8$ is the number of sinc sidelobes included in the summation and η = [-1:1] is the fractional sample offset considered to find the maximum energy. Similarly, the C_{50} and D_{50} can be formulated as follows

$$\mathrm{C}_{\zeta} = 10\log_{10}\left(\frac{\sum_{m=0}^{N_\zeta} h^2(m)}{\sum_{m=N_\zeta+1}^{M-1} h^2(m)}\right)\mathrm{dB}, \tag{2.3}$$

$$\mathrm{D}_{\zeta} = 10\log_{10}\left(\frac{\sum_{m=0}^{N_\zeta} h^2(m)}{\sum_{m=0}^{M-1} h^2(m)}\right)\mathrm{dB}, \tag{2.4}$$

where $\zeta = 50$ ms in this case and N_ζ represents the number of samples in the RIR $h(m)$ from the beginning to ζ ms after the reception of the direct path. Additionally, the Ts is a measure of reverberation that represents the centre of gravity of the squared RIR and it is computed as follows [1]

$$\mathrm{Ts} = \frac{\sum_{m=0}^{M-1} \frac{m}{fs} h^2(m)}{\sum_{m=0}^{M-1} h^2(m)} \ \mathrm{s}, \tag{2.5}$$

where fs is the sampling frequency.

These room acoustic parameters are employed for a wide range of tasks. For example, in [25] a non-linear mapping of T_{60}, DRR and room spectral variance is proposed to estimate the human perception of the reverberation disturbance in speech signals. Kuttruff [1] suggests that D_{50} can be used as an indicator of the speech intelligibility in

reverberant environments. Several room acoustic parameters have been employed to predict the ASR performance for reverberant speech. In [26] a new metric derived from D_{50} is proposed as an estimator of the ASR performance. Tsilfidis et al. [27] present a correlation analysis of several room acoustic parameters (T_{60}, C_{50}, D_{50} ...) showing that C_{50} is the most correlated parameter with ASR performance, reaching the same conclusion as in [12]. In [28] the ASR performance was investigated as a function of early reflection duration. An analysis of the impact of the RIR shape on the ASR performance [29] concludes that the first 50 ms of the RIR barely affect the ASR performance and therefore D_{50} could be used to predict the word accuracy rate. Additionally, several room acoustic parameters have been applied in different dereverberation methods to suppress the reverberation in the signal. C_{50} is used in [13] [9] and T_{60} in [30] [31] to select the ASR acoustic model that better represents the reverberant conditions of the input utterance. In [32] T_{60} is used to add to the current hidden Markov model state the contribution of previous states by applying a piece-wise energy decay curve that is separated in early reflections and late reverberation contributions. The T_{60} information is also applied in [33] to suppress late reverberation through a wavelet packet tree decomposition. From these examples, it is clear that knowledge or estimation of room acoustic parameters can be beneficially exploited in the processing of reverberant signals.

In most real applications, the RIR is unknown and the only available information is the observed reverberant speech signal. Consequently the room acoustic parameters need to be estimated non-intrusively from this signal rather than directly from the RIR. Several methods have been proposed to non-intrusively estimate T_{60}. The method of [34] estimates the decay rate from a statistical model of the sound decay by using the Maximum Likelihood Estimate (MLE) approach and then uses this decay rate to find the MLE estimate for T_{60}. The T_{60} estimator [35] is based on spectral decay distributions. In this case the signal is analysed with a mel-frequency filter bank in order to compute the decay rate by applying a least-square linear fit to the time-frequency log magnitude bins. Variance of the negative gradients in the distribution of decay rates is then mapped to T_{60} with a polynomial function. A method to compute the reverberation time in the modulation domain is proposed in [2], exploiting the fact that low modulation frequency

energy (below 20 Hz) is only slightly affected by the reverberation level whilst high modulation frequency energy increases with the reverberation level. The estimator is created with a Support Vector Regressor (SVR) whose features are the ratio of the average of low modulation frequency energy to different averages of high modulation frequency energy. The overall ratio is then mapped to estimate the DRR. Two methods to estimate T_{60} or C_{80}, which is defined as the clarity index for music [1], from speech and music signals are proposed in [36]. The first method exploits the Power Spectral Density (PSD), which is estimated as the sum of the Hilbert envelopes computed per frequency band. The second method employs a MLE approach to estimate the decay curve of the "cleanest" section in the signal and then averages the partial estimation to create the final estimate. The "cleanest" section is defined as the section with the lowest energy among the free decay phases, i.e. the reverberant tails at the end of words, whose dynamic range is higher than 25 dB. In [37] a multilayer perceptron is built with spectro-temporal modulation features extracted from a 2D-Gabor filter bank in order to estimate the type of room that created the reverberant signal.

Although room acoustic parameters can be also estimated from multi-channel recordings, such as T_{60} [38] or DRR [39], or per frequency bands [40] [41], this chapter focuses on the problem of single-channel full-band room acoustic parameter estimation.

The ACE Challenge [21] provides an extensive database to assess these room acoustic parameter estimators as well as a set of tools to measure their performances which enables to directly compare different methods under the same conditions. The method proposed in this chapter to estimate room acoustic parameters is also evaluated within the ACE Challenge framework in Section 2.6.

2.2 Parameters and evaluation

Before addressing the task of non-intrusive estimation of room acoustic parameters, an analysis of intrusive room acoustic parameters is first performed to investigate the relationship of various room acoustic parameters with ASR performance and thus find the

parameter most correlated with ASR performance.

2.2.1 Room acoustic parameters

The motivation of this work is to estimate the measure of reverberation that is most correlated with the ASR performance. Therefore T_{60}, Ts, DRR, C_ζ and D_ζ over a range of ζ are analysed.

2.2.2 Evaluation metrics

In this context, the ASR performance is measured as the Phoneme Error Rate (PER)

$$\text{PER} = \frac{N_{del} + N_{ins} + N_{sub}}{N_{phn}} \tag{2.6}$$

where N_{phn} is the total number of phonemes in the reference, N_{del} is the number of deletions, N_{sub} is the number of substitutions and N_{ins} the number of insertions. The performance is measured per phoneme to avoid possible influences of the language model or dictionary rules and therefore be able to measure more accurately the impact of reverberation on the acoustic modelling of ASR. For this purpose a context-dependent Gaussian Mixture Models (GMM)-Hidden Markov Models (HMM) phoneme recognizer was employed based on Kaldi [42] following the TIMIT recipe 's5'. The ASR feature vector includes mel-frequency cepstral coefficients with delta and delta-delta features.

In addition to PER, the Perceptual Evaluation of Speech Quality (PESQ) is included in the evaluation as a commonly used metric that is helpful to obtain a quantitative insight into the nature of the test data. PESQ [43] is an intrusive objective method to estimate the speech quality. In this context, the reference signal used in the PESQ calculation is the original anechoic clean speech.

Two different metrics are used to evaluate the relevance of different measures to ASR performance. The first is the absolute value of the Pearson correlation coefficient computed as

$$\rho = \left| \frac{\sum_{u=1}^{N_{utt}} (\beta_u - \overline{\beta})(\alpha_u - \overline{\alpha})}{\sqrt{\sum_{u=1}^{N_{utt}} \left(\beta_u - \overline{\beta}\right)^2 \sum_{u=1}^{N_{utt}} (\alpha_u - \overline{\alpha})^2}} \right|, \tag{2.7}$$

where $\overline{\alpha}$ is the average of the PER scores α_u per utterance, $\overline{\beta}$ is the average of a particular measure of reverberation β_u under consideration computed for each utterance, and N_{utt} is the total number of utterances included. Additionally, the mutual information between these variables computed as [44] is also used

$$I(A;B) = \sum_{\alpha \in A} \int_B p(\alpha, \beta) \log \frac{p(\alpha, \beta)}{p(\alpha)p(\beta)} \, d\beta, \tag{2.8}$$

where the discrete random variable A is the PER and the continuous random variable B is the measure of reverberation, $p(\alpha)$ and $p(\beta)$ are the marginal distribution of A and B respectively and $p(\alpha, \beta)$ is the joint distribution of A and B. The unit of this metric is determined by the base of the logarithm used. In this case the logarithm base 2 is employed and thus the unit is the *bit*. In (2.8) $I(A;B)$ quantifies the reduction in uncertainty about one random variable given another random variable, where the variables in this case are PER scores and the values of a particular measure of reverberation under consideration.

2.2.3 Evaluation data

The data used to compute ρ and $I(A;B)$ for the different measures of reverberation is taken from two sets described in Section 2.4.1.2. The first set is extracted from the training set presented in Section 2.4.1.2 by selecting only the reverberant utterance without noise giving a total of 6144 utterances (5.55 hours). The second set uses the RealInf set from the evaluation set presented in Section 2.4.1.2 which comprises 3960 reverberant utterances (3.70 hours) obtained with measured impulse responses. These two sets comprise different types of RIRs, the former includes only simulated RIRs whereas the latter employs only real measured RIRs, and they are evaluated separately in next section. Besides, no noise is added to the recordings in these sets, thus the reverberation effect on ASR can be more accurately analysed for a wide range of reverberant environments.

2.2.4 Correlation of room acoustic parameters with ASR performance

The correlation and the mutual information of different full-band[2] room acoustic parameters with PER, as well as with PESQ for comparison, is first reviewed in this section. Additionally, the room acoustic parameters computed from each individual mel-frequency subband of the RIR are investigated using the same evaluation metrics.

2.2.4.1 Full frequency-band room acoustic parameters

Table 2.1 displays the correlation coefficients obtained with simulated impulse responses. It shows that the most correlated measure with PER is C_{50}, which is in accordance with the results obtained in [27]. As stated above, the PER is obtained with a context-dependent GMM-HMM phoneme recognizer built with the TIMIT recipe 's5' of Kaldi [42]. Additionally C_{50} is seen again to be the most correlated with PESQ. Figure 2.1 shows the correlation of C_ζ and D_ζ where C_ζ from ζ approximately 20 ms to 50 ms achieves the highest correlation coefficients for PESQ and PER and D_ζ shows its highest correlation coefficients with smaller ζ. Similar results are obtained with measured RIRs which are given in Table 2.2 and in Fig. 2.2.

	T_{60}	DRR	Ts	D_{50}	C_{50}
PER	0.70	0.68	0.73	0.73	**0.85**
PESQ	0.75	0.75	0.78	0.78	**0.91**

Table 2.1: Correlation comparison of PER and PESQ with different acoustic parameters for simulated impulse responses. The maximum values are bold.

	T_{60}	DRR	Ts	D_{50}	C_{50}
PER	0.75	0.37	0.47	0.69	**0.85**
PESQ	0.79	0.42	0.50	0.75	**0.94**

Table 2.2: Correlation comparison of PER and PESQ with different acoustic parameters for real measured impulse responses. The maximum values are bold.

Table 2.3 gives the magnitude of the mutual information between the measure of reverberation and PER and PESQ. It shows that D_{50} and Ts provide the highest mutual

[2]Full-band is used to indicate that the RIR is used without any analysis filter bank being applied.

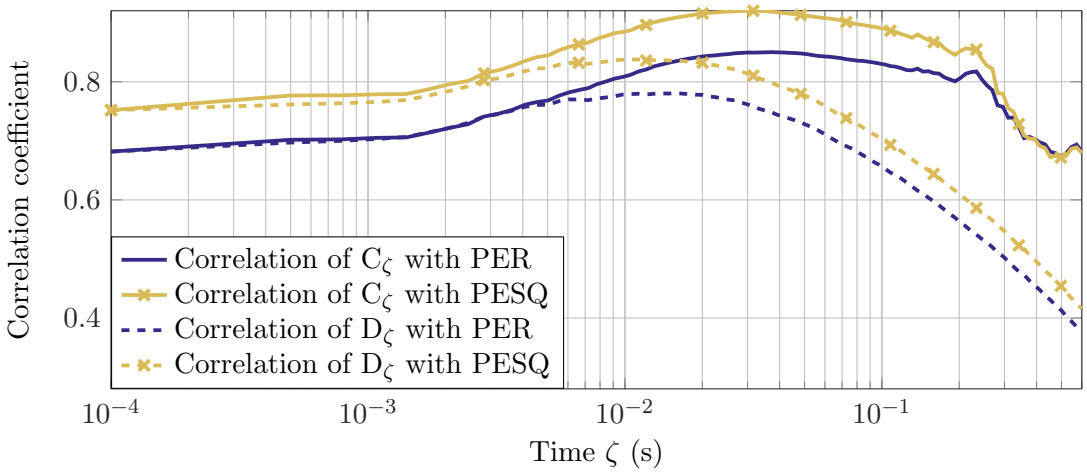

Figure 2.1: PER and PESQ correlation coefficients obtained with C_ζ and D_ζ for ζ between 0.1 ms and 600 ms using simulated RIRs.

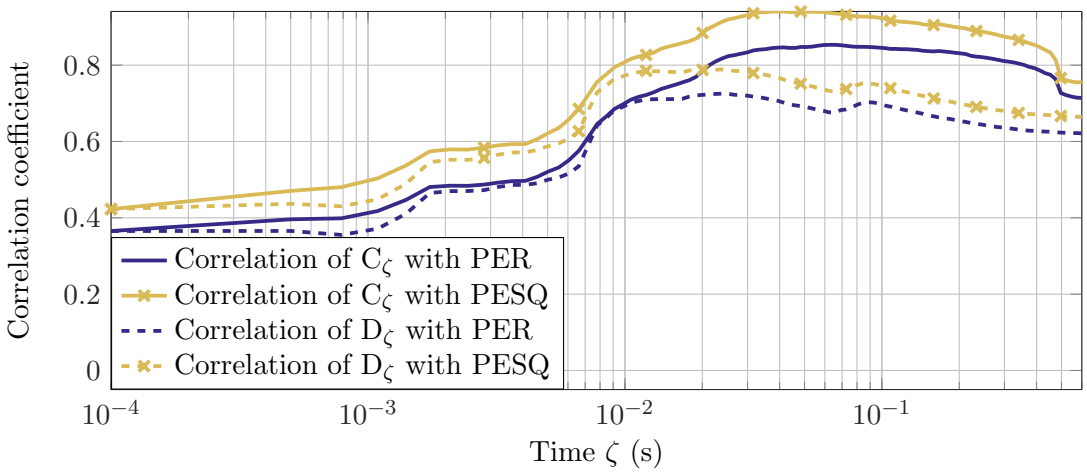

Figure 2.2: PER and PESQ correlation coefficients obtained with C_ζ and D_ζ for ζ between 0.1 ms and 600 ms using real RIRs.

information value with PER and PESQ respectively, closely followed by the C_{50}. DRR is seen to be the measure that shares the least information with PER and PESQ.

Figure 2.3 shows the magnitude of mutual information achieved for C_ζ and D_ζ for a range of ζ from 0.1 ms to 600 ms. It shows similar values for C_ζ and D_ζ. The reason is that C_ζ and D_ζ contain the same information. In fact, setting $\mathcal{X} = \frac{\sum_{m=0}^{N_\zeta} h^2(m)}{\sum_{m=N_\zeta+1}^{M-1} h^2(m)}$, then $C_\zeta = 10\log_{10}(\mathcal{X})$ and $D_\zeta = 10\log_{10}\left(\frac{\mathcal{X}}{1+\mathcal{X}}\right)$, therefore the mutual information is the same for both measures however in Fig. 2.3 the mutual information is not exact the same between C_ζ and D_ζ due to estimation errors computing the mutual information [44]. The

	T_{60}	DRR	Ts	D_{50}	C_{50}
PER	0.71	0.38	0.78	**0.79**	0.75
PESQ	1.12	0.82	**1.26**	1.25	1.23

Table 2.3: Mutual information comparison of PER and PESQ with different acoustic parameters for simulated impulse responses. The maximum values are bold.

highest value of the mutual information with PER is at approximately $\zeta = 50$ ms whereas the highest mutual information values with PESQ are located towards lower ζ values.

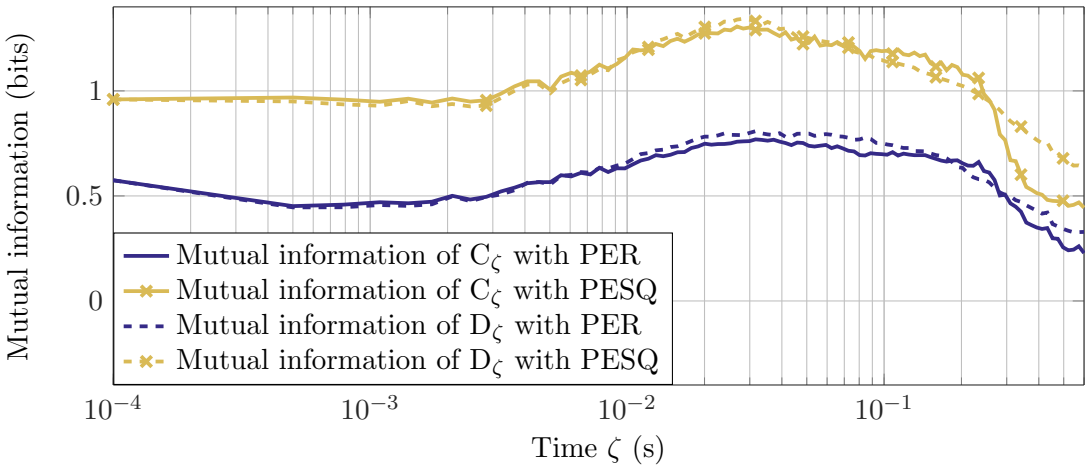

Figure 2.3: PER and PESQ mutual information magnitude obtained with C_ζ and D_ζ for ζ between 0.1 ms and 600 ms using simulated RIRs.

Table 2.4 shows the mutual information magnitude of several measures of reverberation with the ASR performance (PER) and PESQ obtained on reverberant data generated with real measured impulse responses. Despite Ts and T_{60} showing high mutual information in some cases, C_{50} and D_{50} are the measure of reverberation that provides the highest values on average over the two datasets.

Figure 2.4 shows the mutual information of C_ζ and D_ζ with PER and PESQ respectively. All the figures presented in this section lead to the same conclusions: C_ζ provides higher correlation and similar mutual information values compared to D_ζ and the highest values of C_ζ are in the range centred at $\zeta = 50$ ms.

	T_{60}	DRR	Ts	D_{50}	C_{50}
PER	0.79	0.30	0.65	**0.80**	0.76
PESQ	**1.56**	1.11	1.60	1.51	1.46

Table 2.4: Mutual information comparison of PER and PESQ with different acoustic parameters for real measured impulse responses. The maximum values are bold.

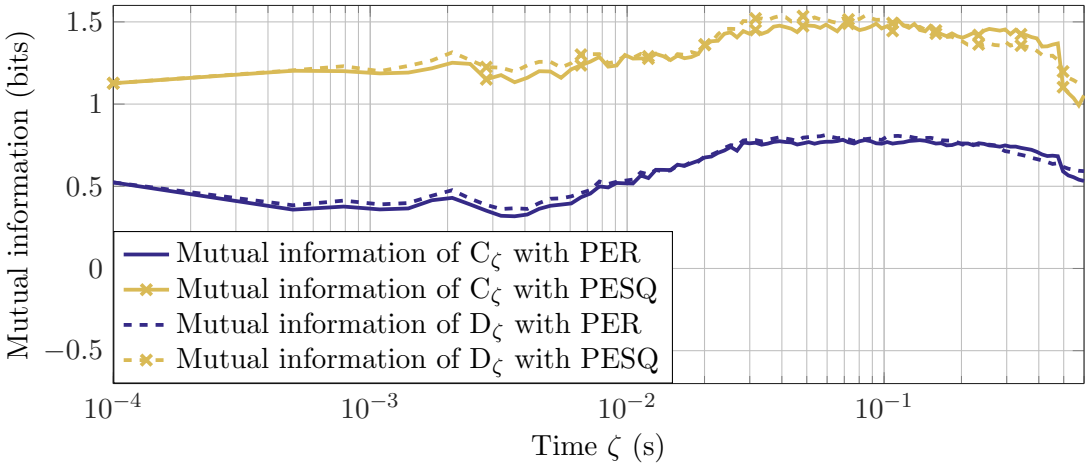

Figure 2.4: PER and PESQ mutual information magnitude obtained with C_ζ and D_ζ for ζ between 0.1 ms and 600 ms using real RIRs.

2.2.4.2 Mel-frequency subbands room acoustic parameters

In ASR, the input acoustic signal is commonly processed to extract the mel-frequency cepstral coefficients [45]. In this section the parameters are computed using the same mel-frequency filter bank applied in the ASR [42] in order to investigate whether room acoustic parameters per mel-frequency subband provide higher correlation and mutual information values than the full-band counterpart. Figure 2.5 illustrates the mel-frequency filter bank response used in this experiment.

Figures 2.6 and 2.7 show the correlation and mutual information values for different acoustic parameters computed per mel-frequency subband for simulated and real impulse responses respectively. The correlation values achieved per mel-frequency subband are lower than (in certain cases approximately equal to) the full-band counterpart, whereas the mutual information computer per mel-frequency subband is in certain bands relatively higher than the full-band value. Thus, not considering combinations of subband or full-band room acoustic parameters, C_{50} computed from the full-band impulse response is the

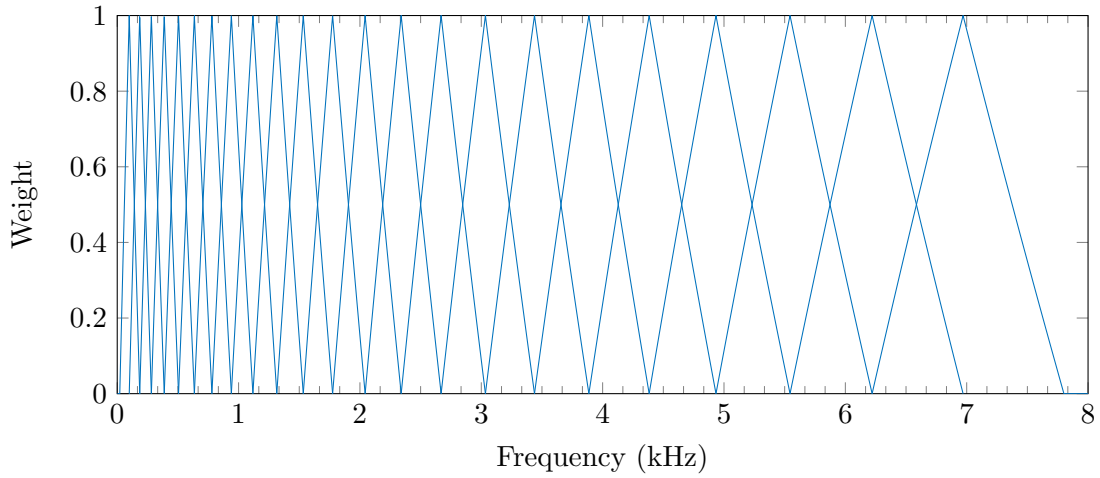

Figure 2.5: Frequency response of the mel-frequency filter bank composed of 23 sub-bands where the lowest frequency is 20 Hz and highest frequency is 7800 Hz.

most correlated room acoustic parameter with ASR performance and provides on average one of the highest mutual information value with ASR performance. Motivated by this finding, in Section 2.3 a method is proposed to estimate full-band C_{50} non-intrusively using only the reverberant speech signal.

The mutual information is bounded between 0 and the minimum entropy of the two random variables under consideration. These upper bounds lie in the range between 5 and 6 bits for all previous experiments which include the calculation of the mutual information. This suggests that the mutual information values achieved in this section are relatively lower than the correlation magnitudes.

2.3 NIRA framework

The proposed method to estimate C_{50} is a data-driven approach which computes 409 features per utterance from a single-channel speech signal at a sampling rate of 8 kHz. Figure 2.8 presents the general block diagram of the NIRA method. The features are used to build a model from which the C_{50} value will be estimated. In this section the attention is focused on estimating C_{50}, however this framework can also be employed to estimate T_{60} and DRR as it is shown in Section 2.6.

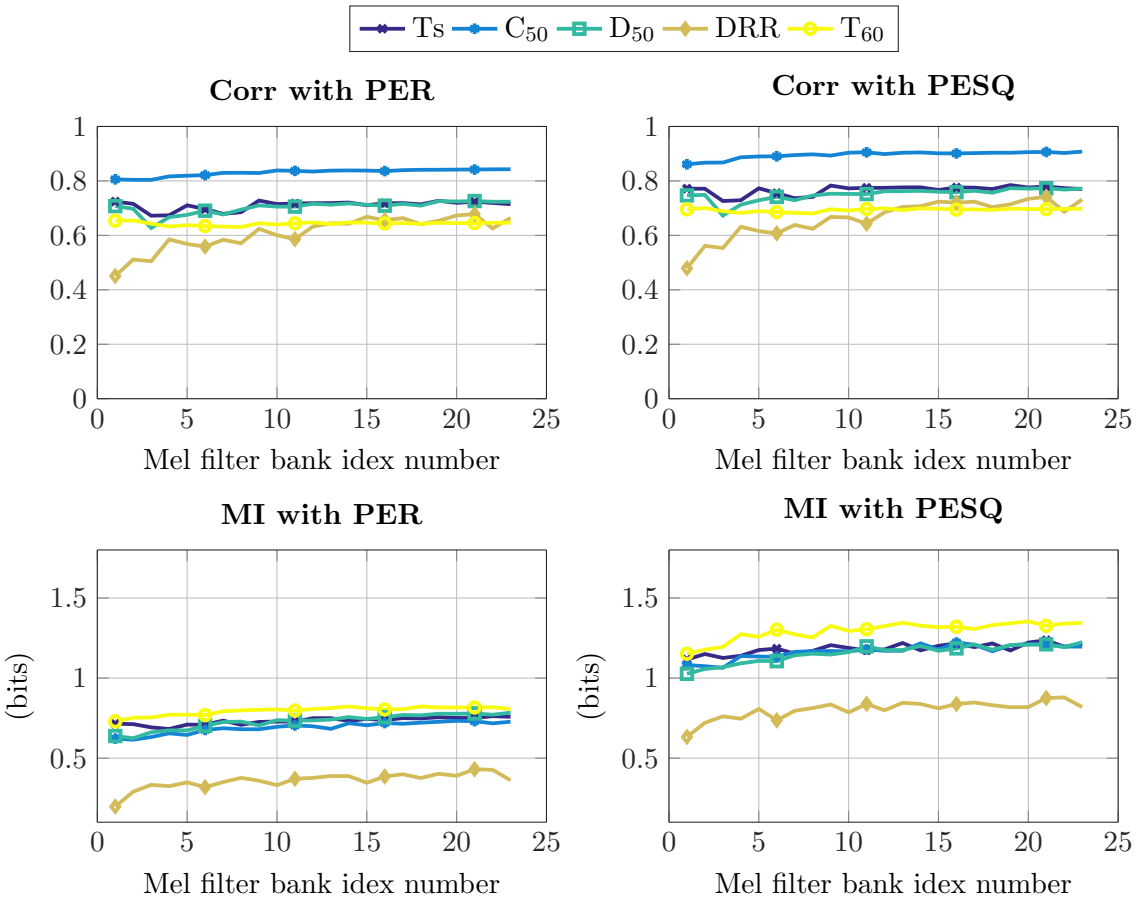

Figure 2.6: PER and PESQ correlation coefficients (top) and mutual information values (bottom) obtained with five measures of reverberation computed per mel-frequency subband using simulated RIRs.

2.3.1 Feature extraction

Features derived from Modulation Domain (MD) [46] and from Deep Scatter Spectrum (DSS) transformation [47] are now proposed.

The modulation frequency representation of speech provides frequency information about long-term speech envelopes. This representation is usually achieved by first computing the Short Time Fourier Transform (STFT) of the input signal and then applying a frequency analysis along the time axis of the STFT with mean normalization. Speech is dominated by modulation frequencies from 2 Hz to 8 Hz [48] which are a key feature in the human auditory system [49]. However, the reverberation effect increases the energy of higher modulation frequencies compared to the energy of lower modulation frequencies [2]

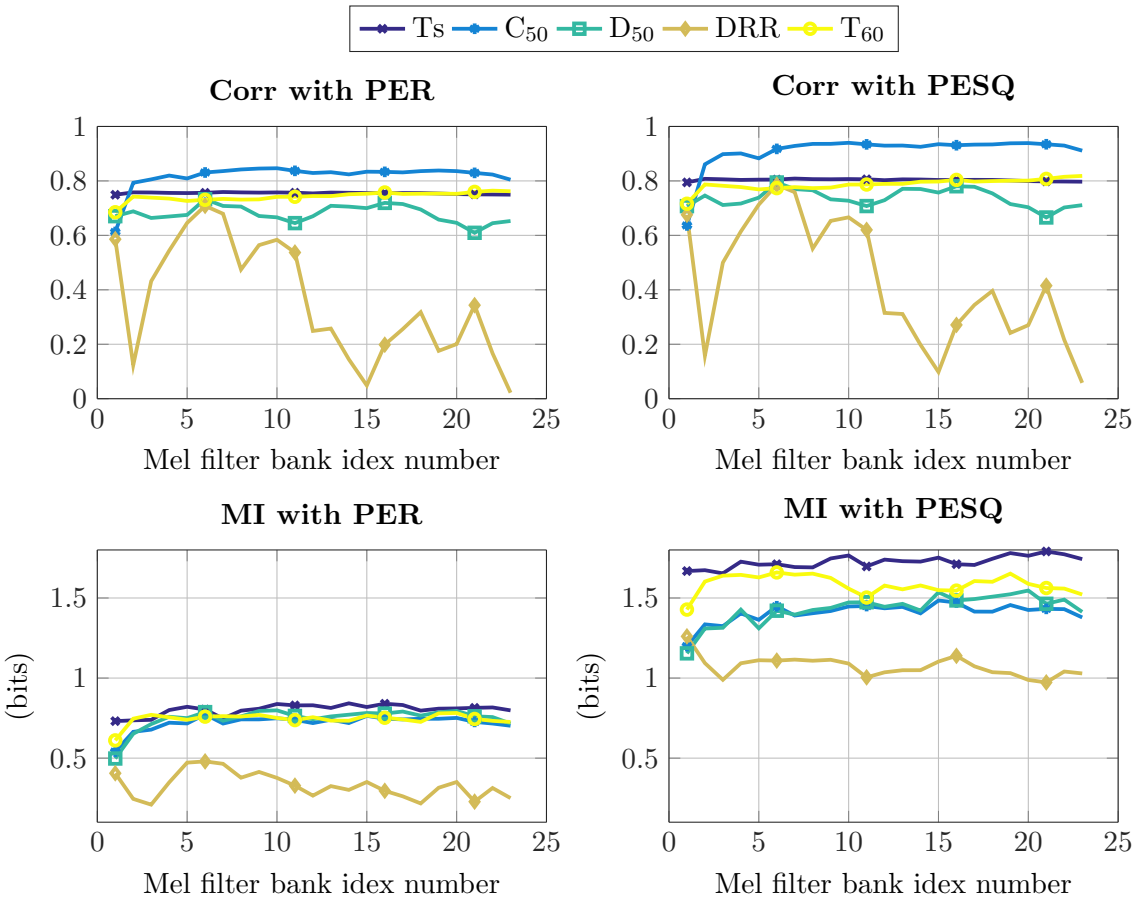

Figure 2.7: PER and PESQ correlation coefficients (top) and mutual information values (bottom) obtained with five measures of reverberation computed per mel-frequency subband using real RIRs.

in the speech signal. Motivated by this fact, modulation domain features are extracted by first selecting the frequency band with the highest energy in the average modulation domain representation and then computing the first four central moments of this frequency band and its two adjacent modulation frequency bands along all acoustic frequency bands.

Deep scattering spectrum features are extracted from a scattering transformation applied to the signal [47]. This wavelet transformation is particularly interesting due to its locally translation invariant representation and its stability to time-warping deformations. The transformation comprises a cascade of wavelet decomposition and modulus operators. The wavelet transformation is created with Morlet constant-Q filter banks. The first-order scattering coefficients are computed following $| y \star \mathcal{T}_1 | \star \mathcal{H}$ where y is the reverberant signal, $\mathcal{T}_1$ represents the wavelets, $\mathcal{H}$ is a low pass filter and $\star$ represents the

convolution operation. The low pass filter is designed such that $\mathcal{T}_1$ and $\mathcal{H}$ cover the whole frequency range of the signal. Similarly, second-order scattering coefficients are calculated from the first wavelet layer output as $|| y \star \mathcal{T}_1 | \star \mathcal{T}_2 | \star \mathcal{H}$ where in this case $\mathcal{T}_2$ represents the wavelets of the second layer. The MFCCs are approximately equal to the first-order scattering coefficients whereas second-order coefficients characterize transient observations (e.g. onsets or amplitude modulation) [47]. Since MFCCs are already included in the feature set and reverberation effect causes distortions in transient periods, features extracted only from this second wavelet layer are employed. The DSS features are computed by using one wavelet per octave in both layers normalized by the first-order coefficients and an average window size of 20 ms with 50% overlap is used.

In addition to these features, the utterance-based and frame-based features inspired by [50] are employed. Utterance-based features are computed from Long Term Average Speech Spectrum (LTASS) deviation by mapping it into 16 bins with equal bandwidth as well as the slope of the unwrapped Hilbert phase of the input signal first proposed in [12]. Unlike frame-based features, utterance-based features provide additional information related to long-term characteristics of the speech utterance which can be suitable to discriminate between different reverberant environments.

Frame-based features comprise the following parameters:

- Line Spectrum Frequency (LSF) features computed by mapping the first 10 Linear Prediction Coefficients (LPC) coefficients to the LSF representation [51].
- Zero-Crossing Rate (ZCR).
- Speech variance.
- Pitch period estimated with the PEFAC algorithm [52].
- Estimation of the importance weighted Signal to Noise Ratio (iSNR) in units of dB [53].
- Variance and dynamic range of the Hilbert envelope [54].

- Three parameters extracted from the Power Spectrum of long term Deviation (PLD): spectral centroid, spectral dynamics and spectral flatness. The PLD is calculated per frame using the log difference between the signal power spectrum and the LTASS power spectrum magnitudes.
- 12th order mean- and variance-normalized MFCCs computed from the fast Fourier transform with delta and delta-delta.

The rate of change for all short-time features, excluding the 12th order MFCCs, is also computed.

A Voice Activity Detector (VAD) is applied to the power-normalized input signal [55], i.e. zero mean and variance equal one, to extract all the features employing only active speech segments. This VAD uses the P.56 method [56].

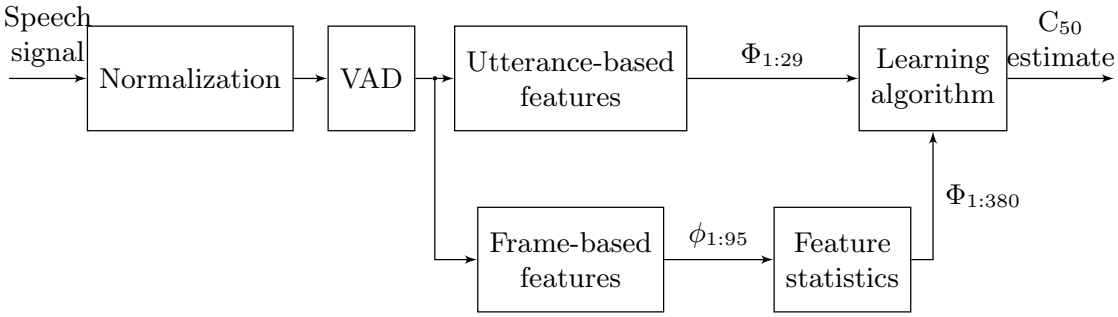

Figure 2.8: The NIRA method.

Table 2.5 summarizes all the features. The complete feature vector is created by appending to the utterance-based features the mean (μ), variance (σ^2), skewness (γ) and kurtosis (κ) of all frame-based features and thereby creating the final vector Φ with 409 features. The importance of these features to estimate C_{50} is analysed in Section 2.4.3.1.

The feature configuration described in Table 2.5 is used to estimate C_{50} per utterance. Additionally, a C_{50} estimated per frame is proposed which employs a different feature configuration. This configuration is based on computing features $\phi_{1:95}$ with a 20 ms window size with 50% overlap and computing $\Phi_{18:29}$ per frame instead of averaging over all per frame modulation domain representations for each utterance. A wider window size with the same overlap is used for the modulation domain features, 256 ms window

Description	Feature	ΔFeature
LSFs	$\phi_{1:10}$	$\phi_{11:20}$
ZCR, Speech variance, Pitch period and iSNR	$\phi_{21:24}$	$\phi_{25:28}$
Variance and dynamic range of Hilbert envelope	$\phi_{29:30}$	$\phi_{31:32}$
Spectral flatness, centroid and dynamics of PLD	$\phi_{33:35}$	$\phi_{36:38}$
MFCCs with delta and delta-delta	$\phi_{39:74}$	-
DSS	$\phi_{75:95}$	-
LTASS	$\Phi_{1:16}$	-
Unwrapped Hilbert phase	Φ_{17}	-
MD	$\Phi_{18:29}$	-

Table 2.5: NIRA features: $\phi_{1:95}$ are frame-based features computed frame by frame, whose statistics are used in the learning algorithm, and $\Phi_{1:29}$ are utterance-based features calculated over the entire utterance. ΔFeature represents the rate of change of the feature.

size, and pitch estimation, 90 ms window size, to preserve higher frequency resolution. The remaining utterance-based features are excluded (i.e. $\Phi_{1:17}$).

2.3.2 Learning algorithms

The learning algorithms employed to build the NIRA models, designed to estimate C_{50} with the features presented in Section 2.3.1, are now presented. The algorithms presented from Section 2.3.2.1 to Section 2.3.2.3 uses the per utterance feature vector configuration, i.e. $\Phi_{1:409}$, whereas Section 2.3.2.4 employs the per frame feature vector configuration, i.e. $\phi_{1:95}$ in addition to the MD per frame features.

2.3.2.1 Classification And Regression Trees (CART)

Classification And Regression Trees (CART) [57] offer a non-parametric methodology to build binary trees. These trees split the data recursively into smaller partitions in order to find the best fit. The training process involves three main steps: tree building, stopping tree building and pruning the tree.

The predicted output is obtained according to the leaf reached after having recursively traversed the tree in depth, deciding the branch to follow at each node based on

one or more input feature values. The CART in a regression mode is used rather than a classification mode since our target is to estimate a room acoustic parameter within a continuous range.

2.3.2.2 Linear regression (LR)

The estimate $\widehat{\mathrm{C}_{50,u}}$ is computed using linear regression [58] as

$$\widehat{\mathrm{C}_{50,\mathrm{u}}} = \sum_{j=1}^{N_{feat}} \vartheta_j \Phi_{j,u} + \vartheta_0, \tag{2.9}$$

where $[\Phi_{1,u}, \ldots, \Phi_{N_{feat},u}]^T$ represents the length-N_{feat} observed variables (i.e. feature vector of length 409) for the uth utterance and $\boldsymbol{\vartheta} = [\vartheta_0, \ldots, \vartheta_{N_{feat}}]$ is a vector comprised of N_{feat}+1 linear regression coefficients.

The optimal coefficient vector $\boldsymbol{\vartheta}$ to model the target $\mathrm{C}_{50,u}$ is obtained by minimizing the sum of squared errors according to the cost function $\mathcal{L}(\boldsymbol{\vartheta})$

$$\mathcal{L}(\boldsymbol{\vartheta}) = \frac{1}{2N_{utt}} \sum_{u=1}^{N_{utt}} \left(\left(\sum_{j=1}^{N_{feat}} \vartheta_j \Phi_{j,u} + \vartheta_0 \right) - \mathrm{C}_{50,u} \right)^2 + \varrho \sum_{j=1}^{N_{feat}} \vartheta_j^2, \tag{2.10}$$

where ϱ is the regularization parameter and N_{utt} represents the total number of utterances. An L2 regularization term is included in the right-hand side of the cost function $\mathcal{L}(\boldsymbol{\vartheta})$ to avoid complex and overfitted models. This minimization problem is solved by applying the gradient descent algorithm [59].

2.3.2.3 Deep Belief neural Network (DBN)

A Deep Belief Network (DBN) structure allows complex non-linear models to learn how to fit the input data to the target C_{50} values. The discriminative training of these networks is applied to a stack of generative pretrained layers. This generative training attempts to learn the structure of the input data in an unsupervised manner by setting the output values to the input values at each layer. Pretrained networks reduce overfitting and discriminative training effort [60]. In this work, sparse autoencoders [61] are used to pretrain

each layer that aim to find optimal weights with the backpropagation algorithm subject to sparsity constraints. This sparsity constraint facilitates the task of finding dependencies on the input data. Additionally, dropout [62] is applied to the discriminative training by randomly removing units of the network at each training step to prevent overfitting. This discriminative training is carried out with stochastic gradient descent and adaptive momentum [63].

Whereas the DBN is widely used for classification tasks, in this work the output layer uses a linear regression on the final hidden layer of neurons in order to estimate a continuous value for C_{50}.

2.3.2.4 Bidirectional Long Short-Term Memory (BLSTM)

RNNs have been applied in different tasks [64] [65] [66] [67]. This type of neural network can be seen as a neural network with at least one feedback connection, hence the output of the activation function is employed to compute the output in the next time step. This configuration provides memory capabilities in the RNN which enables it to learn sequences such as temporal correlations. In addition to the forward propagation, bidirectional RNNs also exploit future context information by processing the data in the time reversed direction. The principal drawback of conventional RNNs is the vanishing gradient problem during learning [68] which is overcome by introducing Long-Short Term Memory (LSTM) cells [69] in the network. LSTM is better at modelling long-term dependencies and it can be combined with a bidirectional RNN to form a bidirectional LSTM.

This structure is employed to build a model [70], which provides a C_{50} estimation per frame, motivated by the bidirectional long-term dependency capabilities of the BLSTM which can potentially represent temporal smearing effects of reverberation.

2.4 NIRA C_{50} estimation

2.4.1 Experimental setup

Experiments have been performed to assess different C_{50} estimators considered in this work. Section 2.4.1.1 defines the evaluation parameters while Section 2.4.1.2 introduces the database employed to evaluate the methods. Section 2.4.2 describes the trained neural network topology finally employed for each model.

2.4.1.1 Evaluation metrics

The Root Mean Square Deviation (RMSD) of C_{50} is computed using

$$E_u = \widehat{C_{50,u}} - C_{50,u} \text{ dB},$$

$$\text{RMSD} = \sqrt{\frac{1}{N_{utt}} \sum_{u=1}^{N_{utt}} (E_u)^2} \text{ dB}, \tag{2.11}$$

where $\widehat{C_{50,u}}$ and $C_{50,u}$, both measured in dB, correspond to the estimated and ground truth C_{50} value of the uth utterance respectively, and N_{utt} is the total number of utterances.

In addition, the mean (μ_E) and standard deviation (σ_E) of the estimation error are also included in the analysis to provide further information about the C_{50} estimation error, and they are computed as

$$\mu_E = \frac{1}{N_{utt}} \sum_{u=1}^{N_{utt}} E_u \text{ dB}, \tag{2.12}$$

$$\sigma_E = \sqrt{\frac{1}{N_{utt}} \sum_{u=1}^{N_{utt}} (E_u - \mu_E)^2} \text{ dB}. \tag{2.13}$$

Moreover, the Pearson correlation coefficient ρ and the mutual information $I(A;B)$ are employed in the analysis to measure the linear relationship between the estimated and ground truth values. They are computed following (2.7) and (2.8), where α_u=$\widehat{C_{50,u}}$, β_u=$C_{50,u}$ and A and B represent the estimated and ground truth C_{50} respectively.

2.4.1.2 Data sets

Three different data sets are employed. The training set is used to train the methods which are tuned with the development set, whereas the evaluation set is used only to evaluate the methods. The utterances, RIRs and noise signals are different for each set and are all sampled at 8 kHz.

2.4.1.2.1 Training set

Speech signals from the TIMIT [71] database are employed to build the training data set. A total of 32 utterances are selected randomly from the TIMIT training set ensuring that 2 different male and 2 different female speakers are included for each dialect while excluding SA sentences. The SA sentences comprise two sentences uttered by all speakers and they are meant to expose the dialectal variants of the speakers. The reverberant speech is created by convolving these speech utterances with simulated room impulse responses. These are randomly generated by using the randomized image method [72] and then RIRs are carefully selected to obtain a set of RIRs with an uniformly distributed C_{50} in the interval [-3,28] dB. A total of 6 RIRs per 1 dB C_{50} band are selected and as a result 192 RIRs are included in this training set. White noise and babble noise from the NOISEX corpus [73] are added to the reverberant speech at Signal-to-Noise Ratio (SNR)s of 0 dB to 30 dB in steps of 5 dB. In realistic environments, the noise needs to be captured in an anechoic environment and convolved as well with a RIR generated in the same room using the same receiver position but a different source position corresponding to the location of the noise source. Due to data limitations, e.g. lack of multiple RIRs in different positions in the room, this noise adaptation is not performed. Instead, the noise recordings are directly added to the reverberant speech. The aim of adding noise to the reverberant speech is to assess the robustness of the estimator.

2.4.1.2.2 Development set

The development set is created following the training set configuration using 16 utterances and 64 RIRs. None of the speech signals nor RIRs of this set are included in the training

set.

2.4.1.2.3 Evaluation set

In the evaluation set, one utterance of each TIMIT core set speaker is included resulting in 24 sentences. The SA sentences are excluded. Babble noise and white noise are also included in the evaluation set at 6 different SNR levels: 2 dB, 7 dB, 12 dB, 17 dB, 22 dB, 27 dB. Both simulated and real measured RIRs are included in this set. Four different databases are considered to build the real room impulse response set: MARDY [3]; REVERB challenge [74]; C4DM RIR [75]; and SMARD [76]. Only recordings from the B-format microphone taken in the Great Hall are considered within the C4DM RIR database due to an artefact in the other C4DM recordings at the 125 Hz octave band. The same selection procedure applied to simulated RIRs is employed in this case to build a set of RIRs with an uniform distribution of C_{50} in the range from -3 dB to 28 dB.

Accordingly, this evaluation set covers a wide range of reverberant scenarios from large rooms such as the Great Hall of the C4DM RIR database to medium rooms with low reverberation as in the SMARD database. Figure 2.9 illustrates the C_{50} distribution of each of the RIR data sets.

The average duration of simulated and real RIRs is 2 s and 1.17 s respectively, i.e. M in (1.1) is on average 16000 for simulated RIRs and 9360 for real RIRs.

In order to provide insights into the performance of the methods for each specific situation, this evaluation set is divided into 26 subsets as outlined in Table 2.6 which are then independently evaluated.

2.4.2 Learning algorithm topologies

The DBN architecture is selected by using Genetic Algorithm (GA) [77] which finds the topology that minimizes the estimation error in the development set. This GA is a heuristic search method which seeks for the optimum solution inspired on the process of natural selection [78]. The GA employs a population of chromosomes which encode potential solutions of the problem. This population is successively replaced with another different

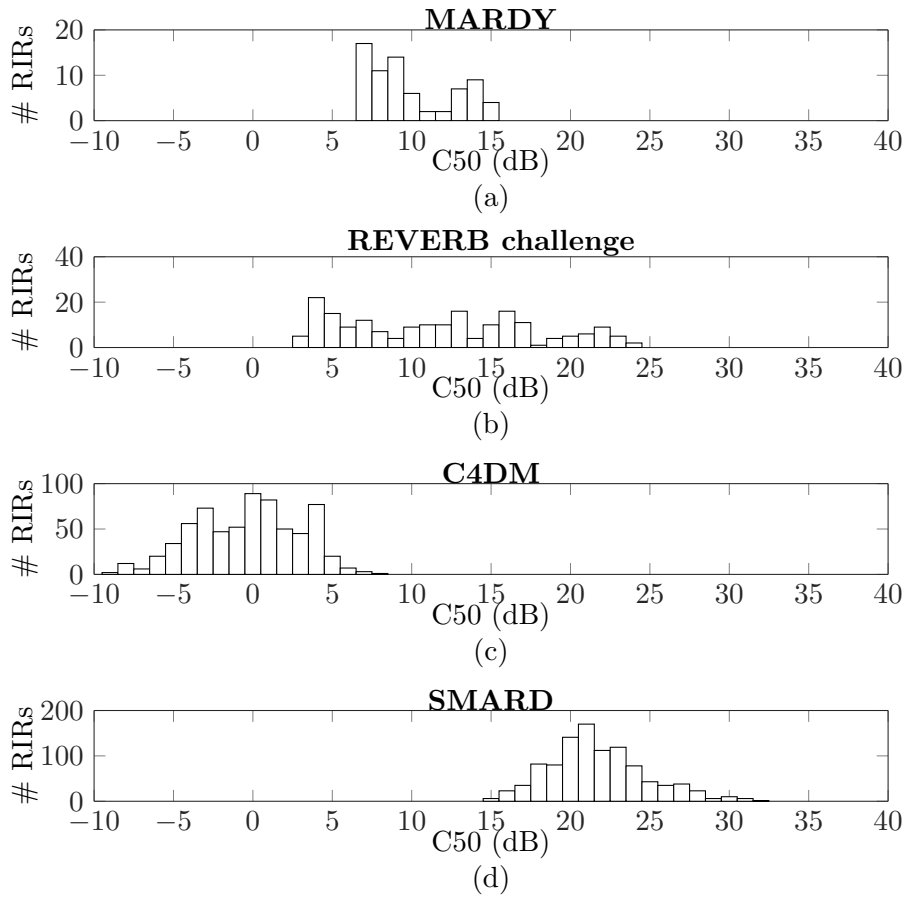

Figure 2.9: Distribution of C_{50} in real measured RIR databases: (a) MARDY database [3]; (b) RIRs collected from the training set of the REVERB challenge database [74]; (c) B-format microphone recording from the Great Hall of the C4DM database [75]; (d) SMARD database [76].

population of chromosomes using three main operations: selection, crossover and mutation. The selection process chooses the chromosomes that are more likely to reproduce according to a fitness function. Then, the crossover operation randomly exchanges two subsequences from two chromosomes to create two new offsprings. Finally, the mutation step randomly changes the value of a chromosome. In this case of finding a optimal DBN topology, the chromosomes encode the network architecture while the fitness function computes the estimation error obtained with the network in the development set. The crossover and mutation probabilities are set to 0.9 and 0.08 respectively. These probabilities were found empirically, nevertheless it was found that small modifications of these parameters lead to approximately the same performance.

RIR type	Noise type	SNR level	Name
Simulated	none	∞	SimInf
	Babble / White	2	SimBA2 / SimWN2
		7	SimBA7 / SimWN7
		12	SimBA12 / SimWN12
		17	SimBA17 / SimWN17
		22	SimBA22 / SimWN22
		27	SimBA27 / SimWN27
Real	none	∞	RealInf
	Babble / White	2	RealBA2 / RealWN2
		7	RealBA7 / RealWN7
		12	RealBA12 / RealWN12
		17	RealBA17 / RealWN17
		22	RealBA22 / RealWN22
		27	RealBA27 / RealWN27

Table 2.6: Subsets of the evaluation set regarding RIR type, noise type and SNR level. In all cases, the same 24 utterances are convolved with 160 RIRs. Therefore each subset comprises 3840 files (approximately 3.6 hours).

Two different CART, Linear Regression (LR) and DBN models are trained for comparison purposes employing different sets of features. Firstly, the feature set proposed in [12] which includes the features $\Phi_{1:17}$ and the statistics (i.e. mean, variance, skewness and kurtosis) of features $\phi_{1:74}$. Secondly, the feature set created with all features presented in Table 2.5. The dimension of the feature vectors are 313 and 409 respectively. The main motivation for this feature vector split is to measure the improvement in performance obtained by including the new features proposed in this work $\Phi_{18:29}$ and the statistics of $\phi_{75:95}$.

The topology selected using GA in the DBN model is a two layer neural network with 75 and 79 neurons in the first and second layer respectively, whereas the model trained with 409 features comprises a first layer of 160 neurons and a second layer of 110 neurons.

The BLSTM model trained with $\phi_{1:74}$ includes 3 layers of 256 neurons in each layer while the model trained with $\phi_{1:95}$ and the 12 MD features computed per frame comprises 4 layers of 64 neurons in each layer.

2.4.3 Performance evaluation

In this section the methods previously presented in Section 2.3 are evaluated. Firstly, in Section 2.4.3.1 an analysis of the importance of the features with respect to the target C_{50} is presented. Two measures of feature importance are used to find the value of the feature to estimate C_{50}. The proposed C_{50} estimators are evaluated in Section 2.4.3.2. A baseline method [36], the only single-channel method found in the literature employed to estimate C_{ζ}, provides a comparison. This baseline method originally estimates C_{80} based on training a neural network and employing PSD features of the reverberant microphone signal. The PSD features are derived from the Welch's periodogram method applied to the sum of the Hilbert envelopes computed per octave band. The PSD is then sampled from 0.15Hz to 25Hz in 1/6 th octave steps, thus a total of 45 features are extracted to provide an C_{80} estimate. However in this case the method has been adapted to estimate C_{50} by modifying the target values in the neural network learning process while preserving the same input features. Finally, the correlation and mutual information values of the C_{50} estimates with ASR performance are compared in Section 2.4.3.3 to an upper bound on the performance, obtained using ground truth C_{50} values.

2.4.3.1 Feature importance

The importance to the C_{50} estimator of each of the features presented in Table 2.5 is now analysed. Numerous methods have been proposed in the literature to compute the feature importance [79] [80]. Two different methods are employed to rank the features according to their importance: CART [57] and Regressional ReliefF method (RReliefF) [81].

The first approach relies on the CART learning algorithm presented in Section 2.3.2.1. This decision tree method attempts to find the feature to split the data set at each node that provides the best discrimination between a set of targets. Once the tree is built, the importance is computed as a function of the purity reduction [57] due to the split at each node. Since CART is employed to estimate C_{50}, the already trained model is also used for feature importance purposes.

The RReliefF [81] method computes the importance of the features based on the capability to differentiate target values that are close together. The importance is defined as a function of three different terms:

- Probability of different feature values given the nearest observations.
- Probability of different target values given the nearest observations.
- Probability of different target values given different feature values and the nearest observations.

This method is used because it provides an importance ranking of the features. Additionally, this method is faster than wrapper methods [82] and it is not targeted to any specific learning algorithm.

Table 2.7 shows the 10 most important features for each method using the features $\phi_{1:74}$ and $\Phi_{1:17}$ proposed in previous work [12]. The ranking of feature importance estimated in each case is different, however there are some common features: ϕ_{29}, ϕ_{52}, ϕ_{64}, ϕ_{65}, ϕ_{66}. The results also suggest that the MFCC features are highly important for C_{50} estimation.

RANK	CART	RReliefF
1	$\sigma^2\phi_{54}$	$\sigma^2\phi_{64}$
2	$\sigma^2\phi_{63}$	$\gamma\phi_{26}$
3	$\mu\phi_{29}$	$\mu\phi_{29}$
4	$\sigma^2\phi_{52}$	$\sigma^2\phi_{66}$
5	$\sigma^2\phi_{64}$	$\mu\phi_{30}$
6	$\sigma^2\phi_{66}$	$\kappa\phi_{26}$
7	$\sigma^2\phi_{28}$	$\gamma\phi_{22}$
8	$\gamma\phi_{52}$	$\sigma^2\phi_{67}$
9	$\sigma^2\phi_{38}$	$\sigma^2\phi_{65}$
10	$\sigma^2\phi_{65}$	$\sigma^2\phi_{52}$

Table 2.7: Ranked feature importance employing CART and RReliefF with the feature set created with $\Phi_{1:17}$ and the statistics of $\phi_{1:74}$ extracted from the training set. The variance, mean, skewness and kurtosis of the per frame features are represented with σ^2, μ, γ and κ respectively.

Table 2.8 shows the top 10 important features for the full feature set, including now the newly proposed MD and DSS features to the previous existing feature set presented

in [12] (i.e. $\phi_{1:74}$ and $\Phi_{1:17}$). CART and RReliefF show some common features to be highly important: Φ_{24} and ϕ_{64}. In both cases, some of the new features (i.e. features within $\phi_{75:95}$ and $\Phi_{18:29}$) are present, in particular MD features appear 8 times in the top 10 for RReliefF. Looking further in the RReliefF ranking (not shown here), DSS features appear 19 times in the first 100 features, which indicates that these features are also important. Additionally, it should be mentioned that CART uses only 46 features after pruning, of which 11 are DSS features and 2 are MD features. These results highlight the suitability of these new features for the estimation of C_{50}.

RANK	CART	RReliefF
1	$\sigma^2\phi_{54}$	Φ_{27}
2	$\sigma^2\phi_{63}$	Φ_{26}
3	Φ_{24}	Φ_{29}
4	$\mu\phi_{29}$	Φ_{19}
5	$\sigma^2\phi_{64}$	$\sigma^2\phi_{64}$
6	$\sigma^2\phi_{66}$	Φ_{18}
7	$\sigma^2\phi_{28}$	$\gamma\phi_{26}$
8	$\sigma^2\phi_{38}$	Φ_{25}
9	$\sigma^2\phi_{118}$	Φ_{21}
10	$\sigma^2\phi_{55}$	Φ_{24}

Table 2.8: Ranked feature importance employing CART and RReliefF with the feature set created with $\Phi_{1:29}$ and the statistics of $\phi_{1:95}$ extracted from the training set. The variance, mean, skewness and kurtosis of the per frame features are represented with σ^2, μ, γ and κ respectively.

2.4.3.2 C_{50} estimators

Figure 2.10 shows a comparison of the estimators' performance with regards to RMSD for all evaluation sets. In this first analysis only features $\Phi_{1:17}$ and the statistics of $\phi_{1:74}$ are included in the feature vector. It is important to note that the BLSTM provides an estimation per frame, hence for comparison purposes only the average of all the frame estimations per utterance is taken into account. Additionally, as mentioned in Section 2.3.2, this learning algorithm employs only the per frame features $\phi_{1:74}$. Figure 2.10 suggests that the estimation accuracy is lower with babble noise compared to the same RIRs with white noise, and estimation accuracy is better in lower levels of noise as expected. The best estimations are achieved with BLSTM, whereas the baseline provides the worst RMSD

values.

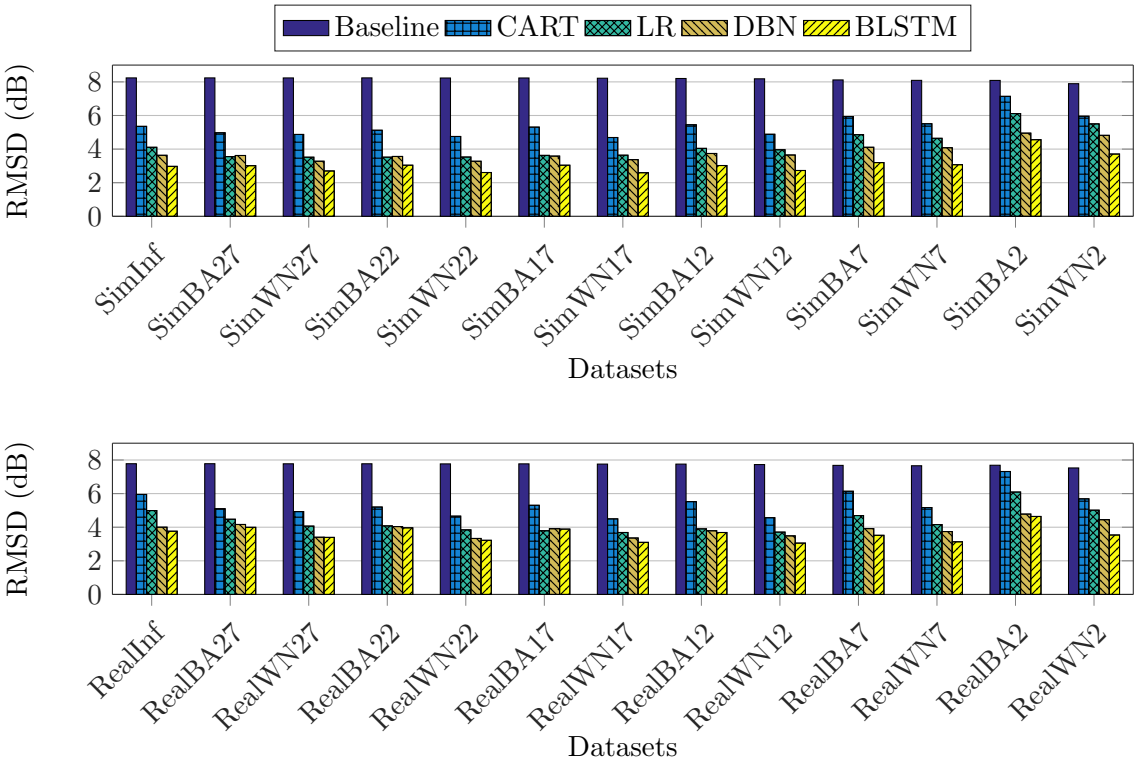

Figure 2.10: RMSD obtained for different room impulse responses (simulated and real) including different noise types (WN: white, BA: babble).

The bias (μ_E) and standard deviation (σ_E) of the estimation errors for each set are shown in Fig. 2.11. CART provides a low-biased estimator. However, due to its high variance the estimation accuracy is degraded. BLSTM achieves the lowest standard deviations for all sets, while the baseline provides the worst bias and standard deviation of the estimation error.

Figure 2.12 plots the improvement in RMSD achieved by including the additional features proposed in this work (i.e. MD and DSS features). This improvement is measured as

$$\Delta\text{RMSD} = \text{RMSD}_w - \text{RMSD}_{w/o}, \tag{2.14}$$

where $\text{RMSD}_{w/o}$ and RMSD_w represent the RMSD obtained without MD and DSS features and with these features respectively. Figure 2.12 shows that almost all estimators improve when using the new features. It is noted that having more features will give an

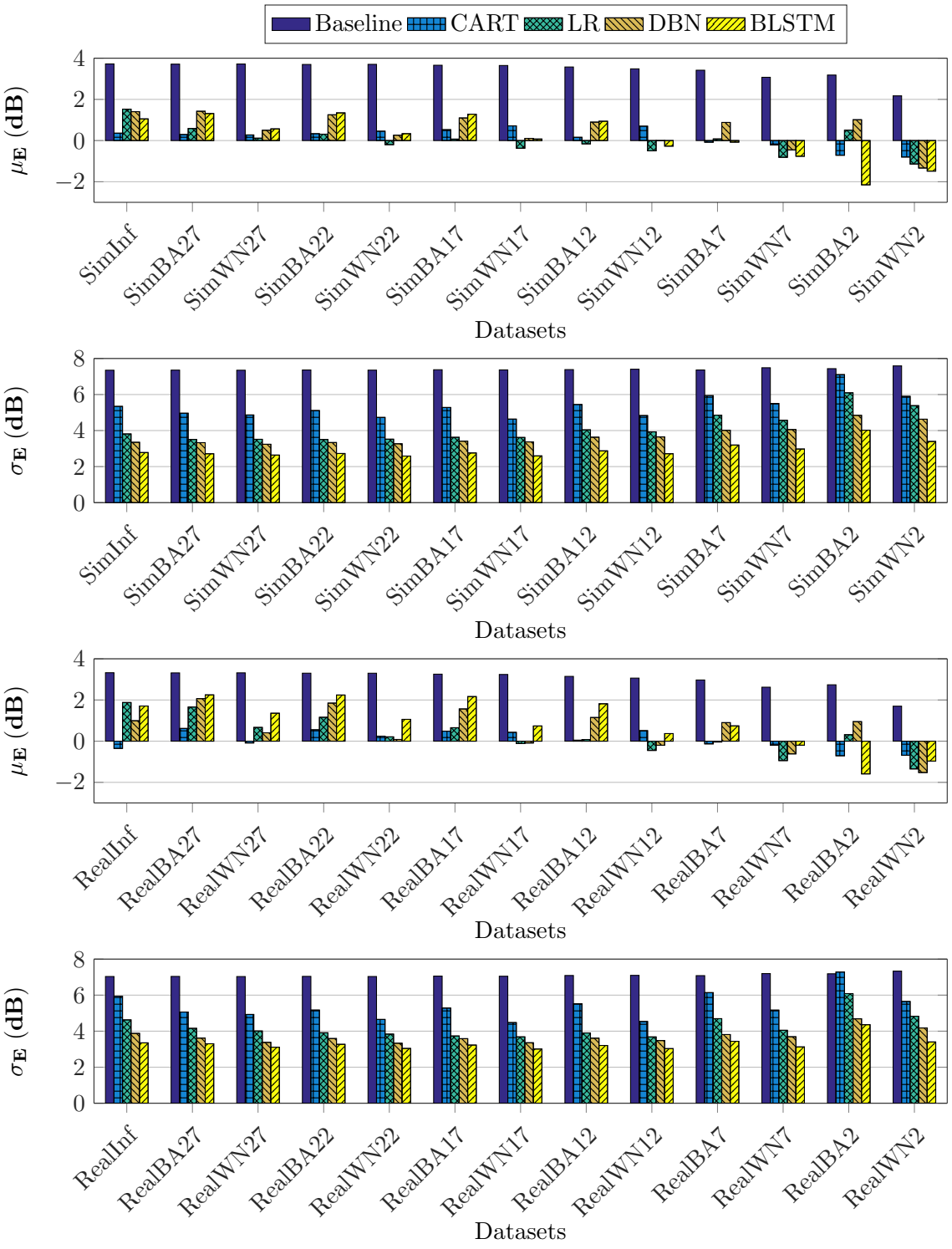

Figure 2.11: Mean and standard deviation of the estimation error obtained for different room impulse responses (simulated and real) including different noise types (WN: white, BA: babble).

improvement when they provide new relevant information however this is not always the case. The highest improvement is achieved with DBN, which is about 0.4 dB on average across all sets. Despite this fact, the best overall performance is achieved with BLSTM, approximately RMSD = 3.3 dB on average.

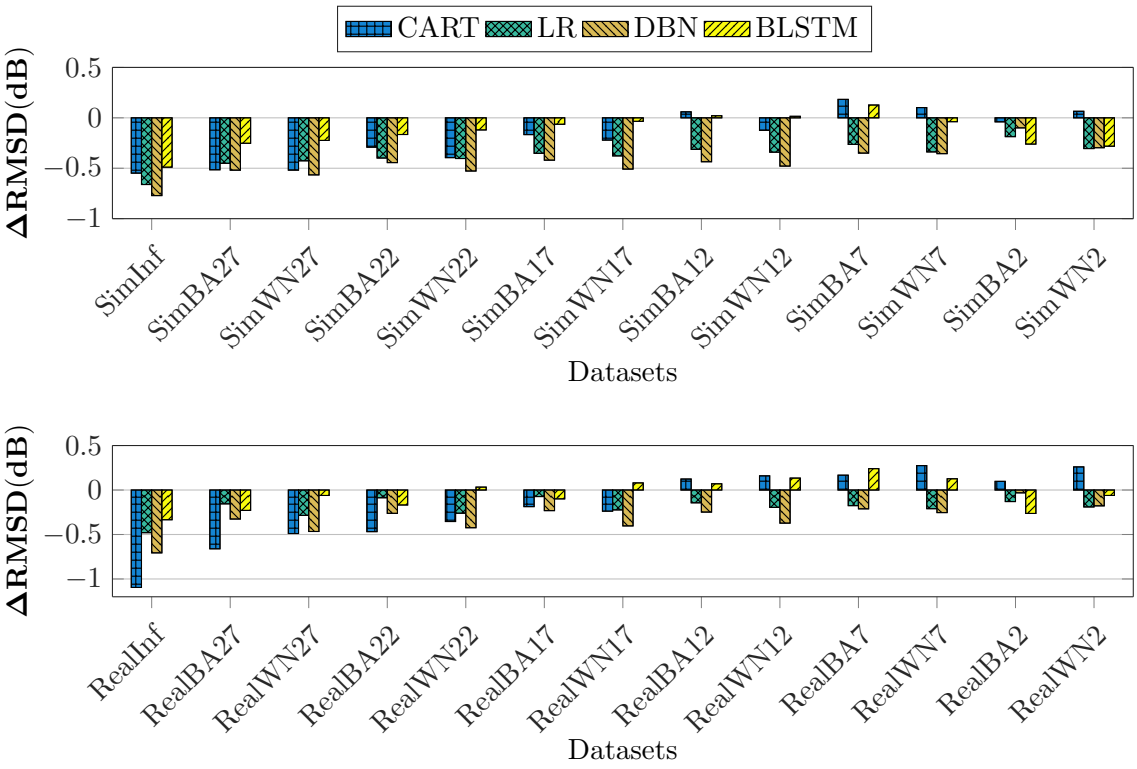

Figure 2.12: RMSD improvement including new features (DSS and MD) for different room impulse responses (simulated and real) including different noise types (WN: white, BA: babble).

Figure 2.13 summarizes the reduction of the bias ($\Delta\mu_E$) and standard deviation ($\Delta\sigma_E$) of the estimation error. These are quantified as follows

$$\begin{aligned} \Delta\mu_E &= \mu_{E_w} - \mu_{E_{w/o}}, \\ \Delta\sigma_E &= \sigma_{E_w} - \sigma_{E_{w/0}}, \end{aligned} \tag{2.15}$$

where the subscripts w/o and w indicate that the C_{50} estimations are obtained without MD and DSS features and with these features respectively. The BLSTM shows a significant reduction of the bias while the standard deviation is increased. On the contrary, all methods except BLSTM achieve a significant reduction of the standard deviation but

their bias is increased.

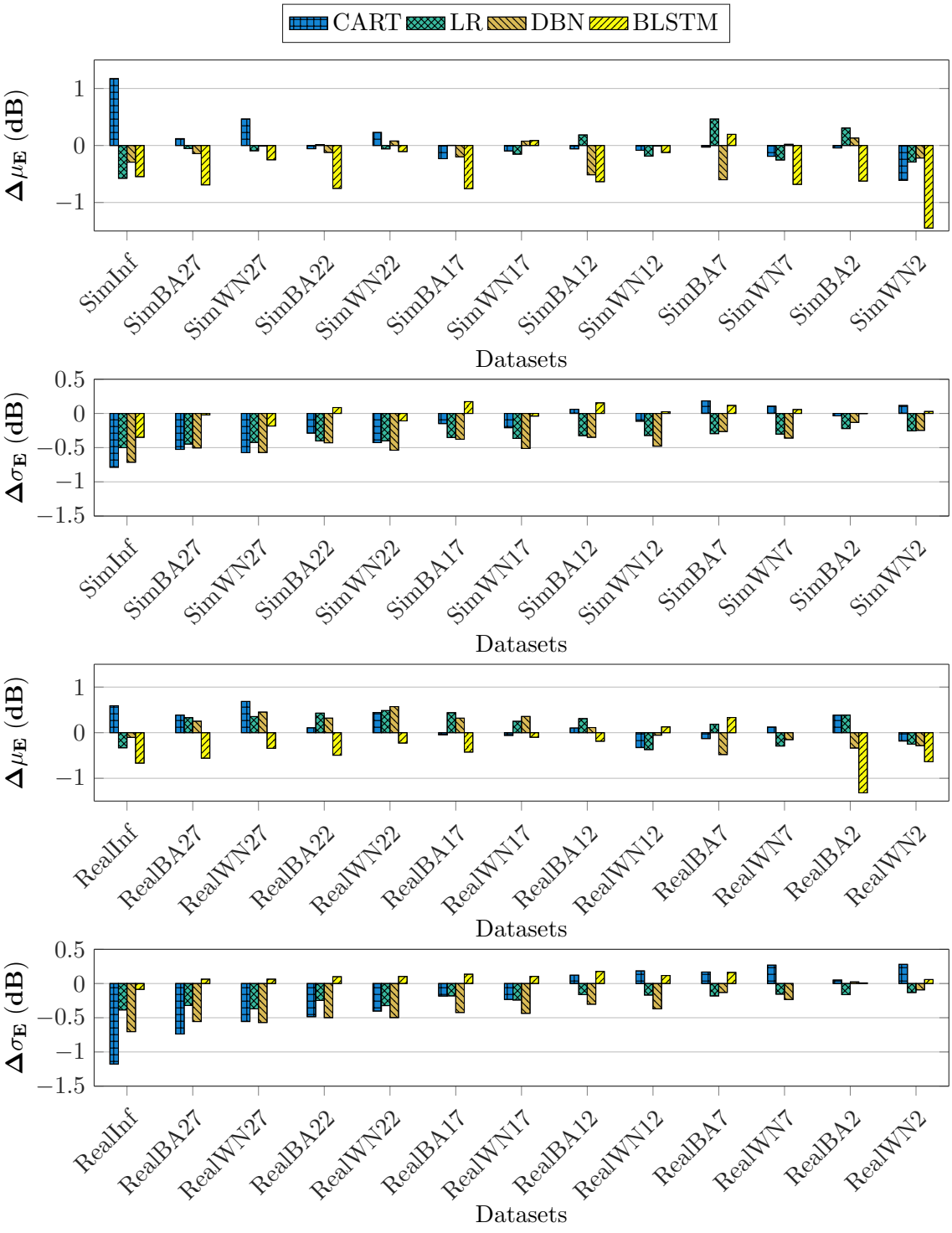

Figure 2.13: Increment of the absolute mean and standard deviation of the estimation error including new features (DSS and MD) for different room impulse responses (simulated and real) including different noise types (WN: white, BA: babble).

Figure 2.14 shows the ground truth C_{50} and the estimated C_{50} for the baseline method and for the BLSTM based method that achieves the lowest RMSD on average. Only two different sets are shown for the sake of clarity: SimInf and SimBA2 which provide approximately the worst and best performance in terms of RMSD for the BLSTM.

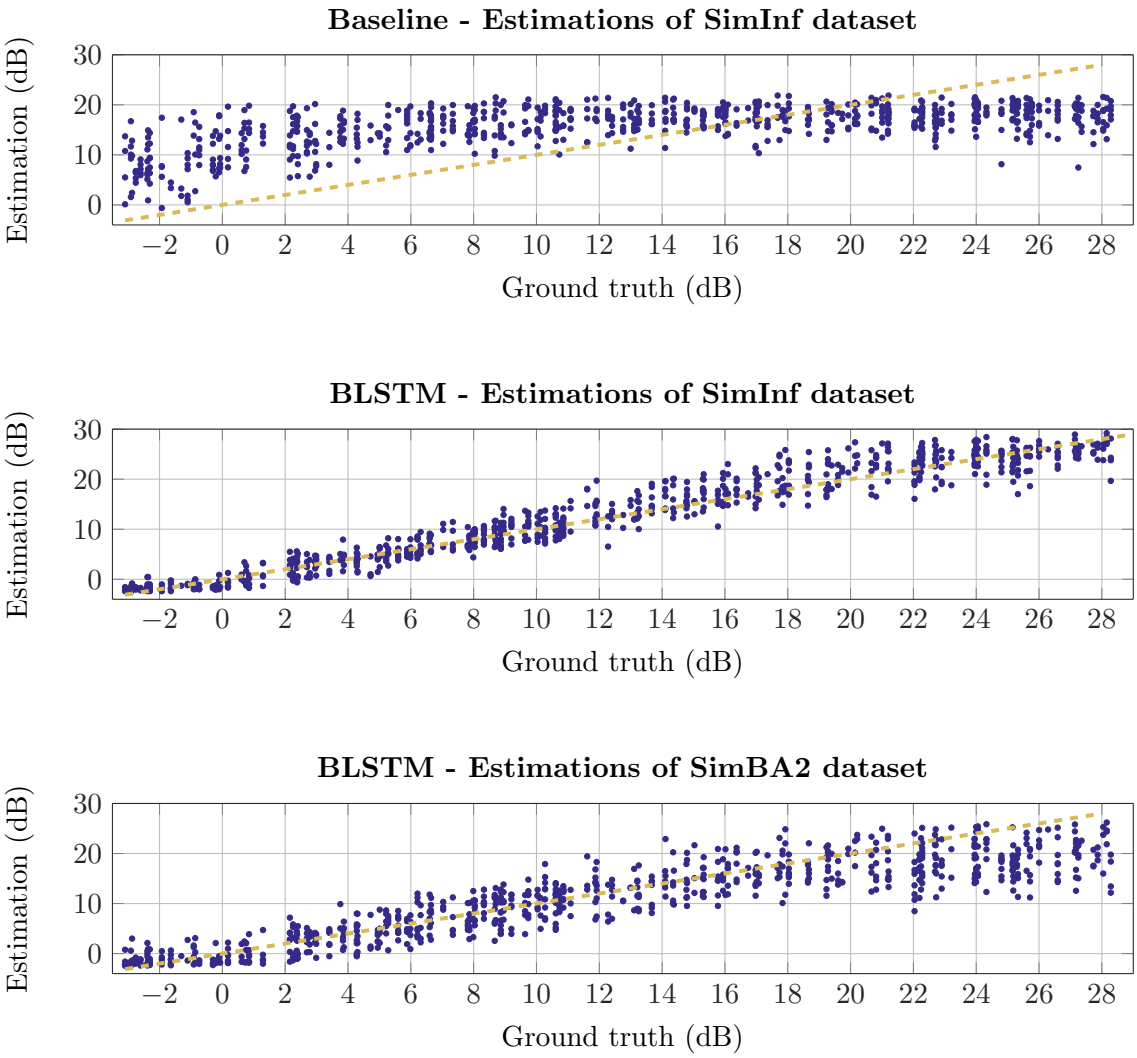

Figure 2.14: Ground truth versus estimated C_{50} of each utterance in SimInf (top) using the baseline method and also in SimInf (middle) and SimBA2 (bottom) evaluation sets employing the BLSTM with all the features, i.e. ϕ_{1-95} and the MD features extracted per frame.

In Fig. 2.15 the C_{50} estimator performance using the BLSTM is analysed per C_{50} bands. In this plot, the width of the C_{50} bands is 3 dB. This figure indicates that NIRA provides more accurate estimations for low C_{50} values, i.e. R1-R3, compared to high C_{50}

values, i.e. R7-R11. In general, RMSD tends to increase as C_{50} increases. Figure 2.15 shows again that in general the accuracy achieved estimating real RIRs is lower than the accuracy obtained for simulated RIRs, even though this is not the case for high C_{50} (R10-R11), and babble noise is again the most problematic condition to estimate C_{50}. Moreover, in environments with high SNR levels, NIRA tends to provide less accurate estimates in middle-high C_{50} bands, i.e. R7 and R8, which is more prominent for real RIRs. This low performance in R7 and R8 bands for real RIRs is mainly due to the spectral characteristics of some SMARD RIRs included in this range which contain strong low frequency components.

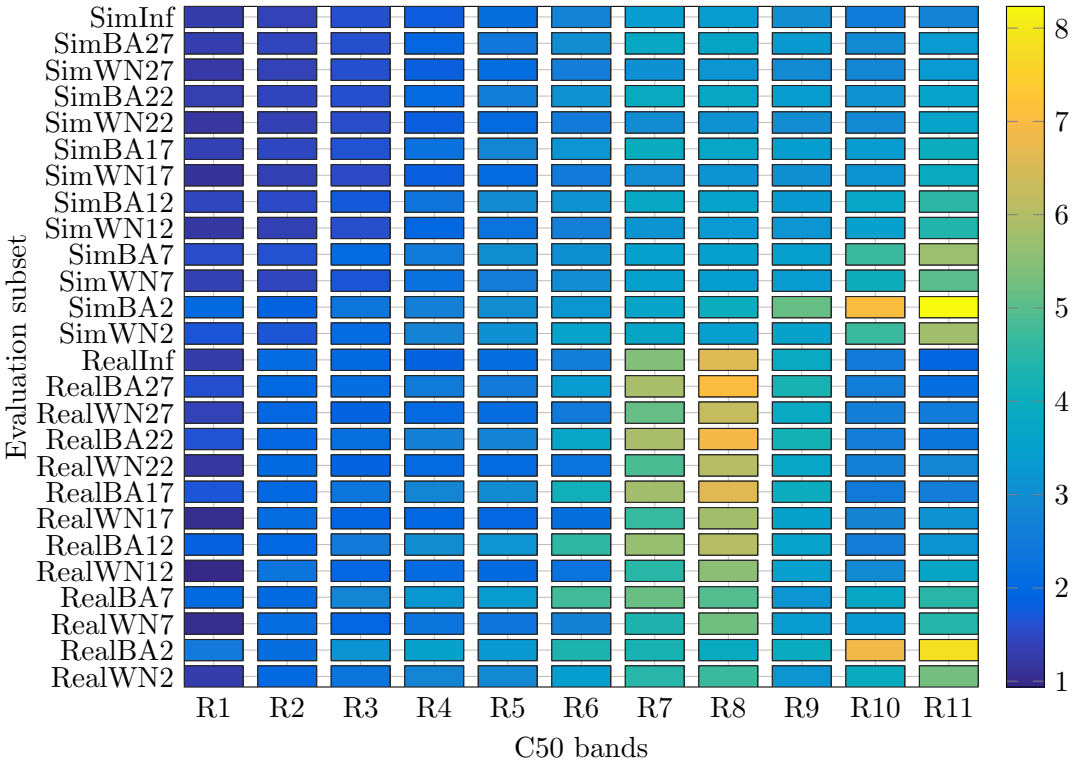

Figure 2.15: Root mean square deviation of the C_{50} estimator for the different evaluation subsets split in different bands according to the ground truth C_{50} (R1: (-4, -1] dB; R2: (-1, 2] dB; R3: (2, 5] dB; R4: (5, 8] dB; R5: (8, 11] dB); R6: (11, 14] dB); R7: (14, 17] dB); R8: (17, 20] dB); R9: (20, 23] dB); R10: (23, 26] dB); R11: (26, 29] dB).

From an application point of view, the minimum number of frames required to provide a C_{50} estimate relatively close to the estimate achieved when using the entire utterance is relevant in order to reduce the computational cost of the estimate and the

latency in real-time applications. For this purpose the per frame performance of the best C_{50} estimator presented previously (i.e. BLSTM) has been analysed. Figure 2.16 illustrates the effect of the number of frames employed to estimate C_{50} on the final RMSD. This performance curve converges to the RMSD value of this estimator, plotted in dashed line in Fig. 2.16, when approximately 180 frames are considered. Taking into account that the window size and increment are 20 ms and 10 ms respectively, approximately 1.9 seconds are required to achieve the same performance as with the full utterance.

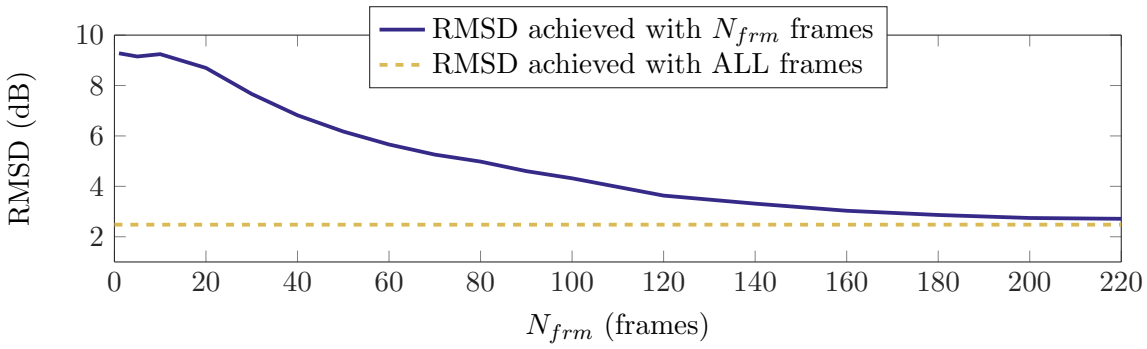

Figure 2.16: RMSD achieved with BLSTM employing the N_{frm} first frames of each utterance in SimInf evaluation set.

Additionally, Fig. 2.17 presents the RMSD calculated with the same estimator when employing N_{frm} frames available for the estimation and averaged per frame l over all the utterances. Note that the RMSD of the frames decreases when N_{frm} increases. The main reason is because BLSTM applies backward propagation from frame N_{frm} to frame l (as well as forward propagation from the first frame to frame l) to provide an estimation at frame l, therefore the performance depends not only on previous frames but on future frames as well. Figure 2.17 indicates that, even from the first frame, a low C_{50} estimate deviation is achieved using 180 frames which is similar to the RMSD obtained with the entire utterance information. Moreover, estimation errors are higher in the first and last frames when N_{frm} is approximately greater than 80 samples which may be due to the fact that the forward propagation for the estimates in the first frames and the backward propagation for the estimates in the last frames only use a limited number of frames to generate the output. This issue creates noisy outputs which are combined with the opposite layer and thus creating a less accurate estimation.

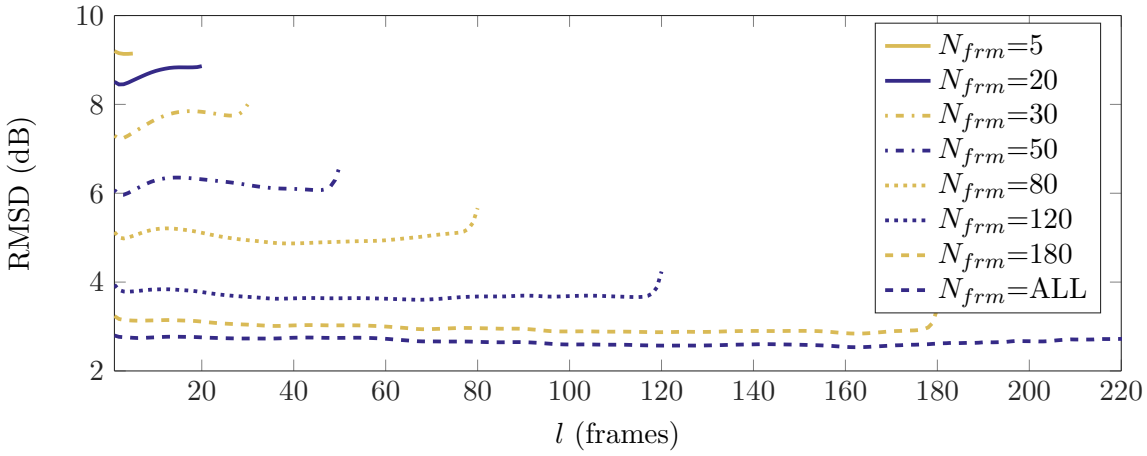

Figure 2.17: RMSD per frame l achieved with BLSTM employing only the N_{frm} first frames of each utterance in SimInf evaluation set to perform the estimation.

2.4.3.3 Correlation and mutual information of the C_{50} estimates with PER

In Section 2.2 it has been shown that ground truth C_{50} values provide high correlation and mutual information values with ASR performance. The correlation and mutual information of the estimated C_{50} values with ASR performance is summarized in Table 2.9. This shows that C_{50} estimates provide a high correlation value which is comparable to the value obtained with the ground truth C_{50} values. Furthermore, the use of C_{50} within the context of speech recognition is investigated in Section 3.4.

Metric	GT	Baseline	CART	LR	DBN	BLSTM
ρ	0.85	0.56	0.77	0.84	0.85	0.85
$I(A;B)$	0.66	0.36	0.67	0.67	0.69	0.73

Table 2.9: Correlation (ρ) and mutual information ($I(A;B)$) values of the ground truth C_{50} (GT) and the estimated C_{50} (Baseline, CART, LR, DBN and BLSTM) with PER for RealInf evaluation set.

2.4.4 Conclusions

It has been shown that the full frequency-band C_{50} is the most relevant measure of reverberation to predict phoneme recognition in terms of correlation and mutual information. Motivated by this finding, a data-driven method (NIRA) has been proposed to estimate C_{50} from the reverberant speech signal using a single-channel microphone signal. Four

different approaches have been evaluated as well as the importance of the input features. The approach that employs BLSTM has shown the best performance on average across all evaluation sets, which include measured impulse responses, achieving a root mean square deviation of 3.3 dB in C_{50} estimation. This deviation is similar to the minimum C_{50} variation necessary to perceive a change in reverberant speech in everyday situations stated in [83] to be in the region of 3 dB.

2.5 NIRA C_{50} prediction intervals and confidence measures

Estimates of room acoustics parameters have a number of applications as it was shown in Section 2.1. In addition, information related to the accuracy of the estimator can be important in many situations in order to quantify the risk of applying the estimate to an application. In this section two methods that provide this type of information are explored. The first one is based on prediction intervals and the second one is a confidence measure.

For the sake of clarity, we describe in the following bullet points four different C_{50} notations employed throughout this paper to refer to different approaches of computing C_{50}:

- $C_{50,u}$ is the C_{50} computed directly from the RIR used to create the reverberant signal y_u employing (2.3).
- $C_{50,u}(y_u)$ is the hypothesized C_{50} in the reverberant signal y_u. This is effectively the $C_{50,u}$ with the speech spectrum bias due to spectral coloration of y_u and it is employed to described the prediction intervals in Section 2.5.1.
- $\widehat{C_{50,l,u}}(y_u)$ is the C_{50} estimated at frame l from the reverberant signal y_u which is obtained with NIRA method described in Section 2.4.
- $\widehat{C_{50,u}}(y_u)$ is the C_{50} estimated per utterance from the reverberant signal y_u. This estimate is computed as the average of the per frame estimates $\widehat{C_{50,l,u}}(y_u)$ for the given reverberant signal y_u.

2.5.1 Prediction intervals

Prediction intervals provide an indication of the accuracy of the C_{50} estimates. These intervals are defined with an upper and lower bound and their values are unbounded and related to a C_{50} range.

In practice, the ground truth $C_{50,u}$ estimate, computed directly from the RIR may differ from the hypothesized $C_{50,u}(y_u)$ in the reverberant signal. This difference is caused by several factors. One of these is due to the spectrum of the speech signal y_u which is colored and therefore only some frequencies of the RIR are excited in the reverberant signal. Therefore $C_{50,u}(y_u)$ will differ by $\epsilon_u(y_u)$ from $C_{50,u}$ and can be written as follows,

$$C_{50,u} = C_{50,u}(y_u) + \epsilon_u(y_u) \quad \text{dB}, \tag{2.16}$$

Figure 2.18 shows boxplot of the error $\epsilon_u(y_u)$ obtained for the 24 different TIMIT utterances included in the evaluation set and employing one recorded RIR from MARDY database [3]. This error is computed as the average of the per frame errors where $C_{50,u}(y_u)$ per frame is computed using (2.3) from the RIR generated after convolving the given speech frame with the original recorded RIR. The frame size employed is 20 ms with an 50% overlap and the frames containing silence are discarded. The magnitude of the error $\epsilon_u(y_u)$ obtained is on average 1.33 dB.

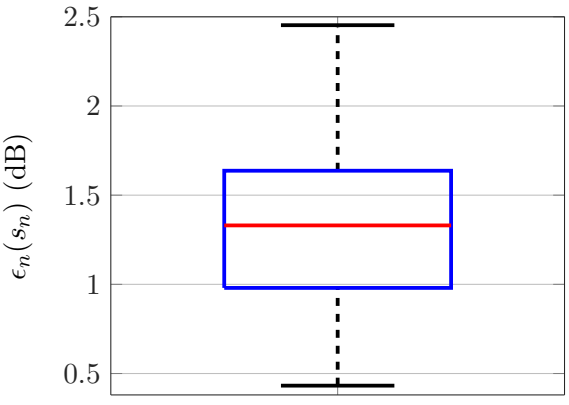

Figure 2.18: Boxplot of the $\epsilon_u(y_u)$ obtained with different utterance y_u using the same RIR.

The equation (2.16) can be rewritten as,

$$C_{50,u} - \widehat{C_{50,u}}(y_u) = (C_{50,u}(y_u) - \widehat{C_{50,u}}(y_u)) + \epsilon_u(y_u) \text{ dB}. \tag{2.17}$$

While confidence intervals address the differences between $C_{50,u}(y_u)$ and $\widehat{C_{50,u}}(y_u)$, prediction intervals deal with the left-hand side of (2.17), which is related to $p(C_{50,u} \mid \widehat{C_{50,u}}(y_u))$ [84]. The latter parameter is used since the estimator is evaluated using the ground truth $C_{50,u}$.

There are two sources of error in (2.17): first the standard deviation Ξ_v due to data limitations, e.g. the spectrum is colored, which is associated to the error $\epsilon_u(y_u)$, and second the standard deviation Ξ_m due to model limitations causing estimation errors associated with the magnitude $C_{50,u}(y_u) - \widehat{C_{50,u}}(y_u)$. Considering these errors to be statistically uncorrelated, we can write

$$\Xi_t = \Xi_v + \Xi_m \text{ dB}. \tag{2.18}$$

In order to test this assumption, the correlation between these two sources of error is measured using the data employed above to obtain the magnitude of $\epsilon_u(y_u)$ shown in Fig. 2.18. The correlation coefficients are computed per utterance using the errors obtained per frame. The average of these coefficients is 0.07, which indicates a weak correlation between Ξ_v and Ξ_m suggesting that (2.18) is a valid assumption.

Confidence intervals are based on the standard deviation Ξ_m, however prediction intervals are based on Ξ_t [85].

In this section it is proposed to estimate Ξ_t for the uth utterance ($\widehat{\Xi_{t,u}}$) from the N_{frm} per frame estimates $\widehat{C_{50,l,u}}(y_u)$ as

$$\widehat{\Xi_{t,u}} = \sqrt{\frac{1}{N_{frm}} \sum_{l=1}^{N_{frm}} (\widehat{C_{50,l,u}}(y_u) - \widehat{C_{50,u}}(y_u))^2} \text{ dB}.$$

where

$$\widehat{\mathrm{C}_{50,u}}(y_u) = \frac{1}{N_{frm}} \sum_{l=1}^{N_{frm}} \widehat{\mathrm{C}_{50,l,u}}(y_u)) \quad \text{dB}.$$

The lower ($\Omega_{low,u}$) and upper ($\Omega_{up,u}$) bounds of the Prediction Interval (PI) for the uth utterance are computed as

$$\Omega_{low,u} = \widehat{\mathrm{C}_{50,u}}(y_u) - \mathcal{K} \cdot \widehat{\Xi_{t,u}} \quad \text{dB}, \tag{2.19}$$

$$\Omega_{up,u} = \widehat{\mathrm{C}_{50,u}}(y_u) + \mathcal{K} \cdot \widehat{\Xi_{t,u}} \quad \text{dB}, \tag{2.20}$$

where $\mathcal{K}$ is a tuning parameter that defines the width of the intervals. Figure 2.19 shows different PI computed for a given $\widehat{\mathrm{C}_{50,l,u}}(y_u)$ using several values of $\mathcal{K}$.

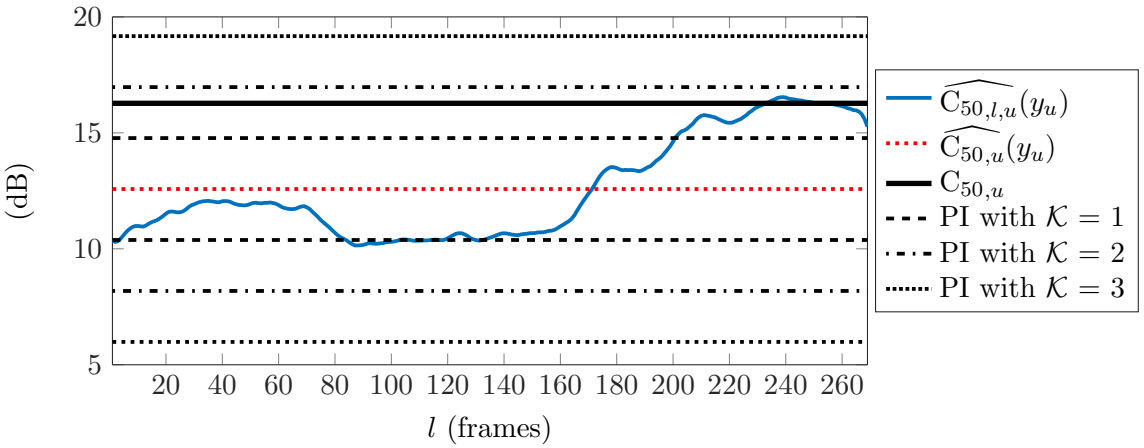

Figure 2.19: Different PIs depending on $\mathcal{K}$ for one utterance of the development set.

2.5.2 Confidence measure

In this section, a confidence measure which provides a metric bounded from 0 to 1, unlike prediction intervals, and related to the C_{50} estimation performance is proposed. This confidence measure is based on normalizing the prediction interval width, $\Omega_{up,u} - \Omega_{low,u}$, by the total C_{50} range observed in the training dataset, R_{tr}. Thus, the confidence measure CM_u for the uth utterance is computed as

$$\mathrm{CM}_u = \max\left\{1 - \frac{\Omega_{up,u} - \Omega_{low,u}}{\mathrm{R}_{tr}}, 0\right\}, \tag{2.21}$$

where $\Omega_{up,u} - \Omega_{low,u} = 2 \cdot \mathcal{K} \cdot \Xi_t$. The normalized width $(\Omega_{up,u} - \Omega_{low,u})$ / R_{tr} is subtracted from 1 in order to make the confidence measure directly proportional to the accuracy level and the maximization is carried out to ensure that the lower bound of the confidence measure is 0.

2.5.3 Experimental setup

The data sets employed to evaluated the prediction intervals and the confidence measure are described in Section 2.4.1.2.

2.5.3.1 Evaluation metrics

The estimated prediction intervals are evaluated using the Prediction Interval Coverage Probability (PICP) and the Normalized Mean Prediction Interval Width (NMPIW) metrics [84]. The PICP measures the percentage of times the ground truth $C_{50,u}$ lies between the upper and lower bound of the estimated PI:

$$\text{PICP} = \frac{1}{N_{utt}} \sum_{u=1}^{N_{utt}} c_u \cdot 100 \quad \%,$$

where

$$c_u = \begin{cases} 1, & C_{50,u} \in [\Omega_{low,u}, \Omega_{up,u}], \\ 0, & C_{50,u} \notin [\Omega_{low,u}, \Omega_{up,u}]. \end{cases}$$

The PICP provides information about the accuracy of the PI, however additional information regarding the width of the PIs is needed since high PICP can be achieved by using wide intervals. The width of the intervals is measured using NMPIW:

$$\text{NMPIW} = \frac{\frac{1}{N_{utt}} \sum_{u=1}^{N_{utt}} (\Omega_{up,u} - \Omega_{low,u})}{R_{eval}} \cdot 100 \quad \%,$$

where R_{eval} represents the range of $C_{50,u}$ values observed in the evaluation set.

Finally, the confidence measure method is evaluated using the Pearson correlation between the absolute C_{50} estimation error $|C_{50,u} - \widehat{C_{50,u}}(y_u)|$ and the confidence values

CM_u.

2.5.4 Results

The NIRA model used for this evaluation is the same BLSTM configuration with all features, i.e. $\phi_{1:95}$ in addition to the MD per frame features, assessed in Section 2.4.3.

2.5.4.1 Prediction intervals

Figure 2.20 shows the PICP and the NMPIW for the development set using different values of $\mathcal{K}$. The PICP rapidly increases with $\mathcal{K}$ for low values of $\mathcal{K}$, whereas NMPIW has a linear behaviour in the whole range of $\mathcal{K}$. Assuming a high PICP is required for a given application, e.g PICP = 80%, this is achieved, as shown in Fig. 2.20, at $\mathcal{K} = 5.6$ with a NMPIW = 30.20% in the development set. It is worth noting that NMPIW = 100% may not indicate that prediction intervals cover the full ground truth range [-3,28] dB, it only suggests that the width of the intervals is comparable to the width of the ground truth C_{50}. In contrast, if the estimate $\widehat{C_{50,u}}(y_u)$ is in the middle ground truth range, i.e. $\widehat{C_{50,u}}(y_u) = 15.5$ dB, the prediction interval bounds are the same as the ground truth range limits when NMPIW = 100%.

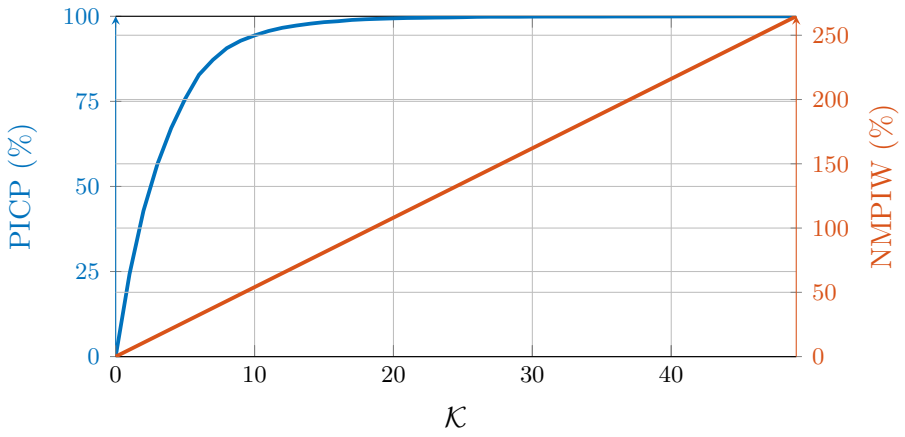

Figure 2.20: Values of PICP and NMPIW depending on the tuning parameter $\mathcal{K}$ tested on the development set.

The evaluation set is used in order to assess the repeatability of this performance. Figure 2.21 shows the PICP and NMPIW absolute differences between development set

and each evaluation subset for $\mathcal{K} = 5.6$. It shows higher PICP for many of the subsets that include simulated RIRs with limited variations of NMPIW, and slightly lower PICP for many of the subsets that include real RIRs. The main reason for this difference is due to the characteristics of the training set which only includes simulated RIRs. The motivation for using only this type of RIRs is that simulated RIRs using the randomized image method can be obtained faster and with less resources than real RIRs which need to be recorded in a given room. Moreover room dimensions and reverberation time can be directly configured with the randomized image method. However, this method makes some assumptions such as perfect rectangular geometry of the rooms, homogeneous propagation medium or specular sound reflections which are not realistic. Due to these limitations, the RIRs generated may have different characteristics compared to real RIRs, e.g. different reflection densities in the late reverberation, affecting the reverberation impact on the original utterance and hence affecting also the estimation of C_{50}.

The higher NMPIW values for the scenarios which include babble noise indicate that this type of noise creates higher variations in the per frame estimate $\widehat{C_{50,l,u}}(y_u)$ compared to the other subsets.

Overall, Figure 2.21 shows that the values of PICP and NMPIW for each evaluation subset are consistent with the PICP and NMPIW obtained in the development set.

2.5.4.2 Confidence measure

As it was shown in the confidence measure formulation (2.21), this metric depends on the value of $\mathcal{K}$. In this section, all the results are obtained with the same value of $\mathcal{K}$ applied to the prediction intervals example, i.e. $\mathcal{K} = 5.6$.

Figure 2.22 shows the confidence values obtained in the development set. In addition to the individual confidence values, the conditional average extracted from these points is plotted. The conditional average is computed by averaging the absolute estimation error differences with similar confidence measure values. This average is carried out over confidence measure bands obtained after sorting all the points by confidence value and then splitting these points in 10 bands which comprise the same number of elements.

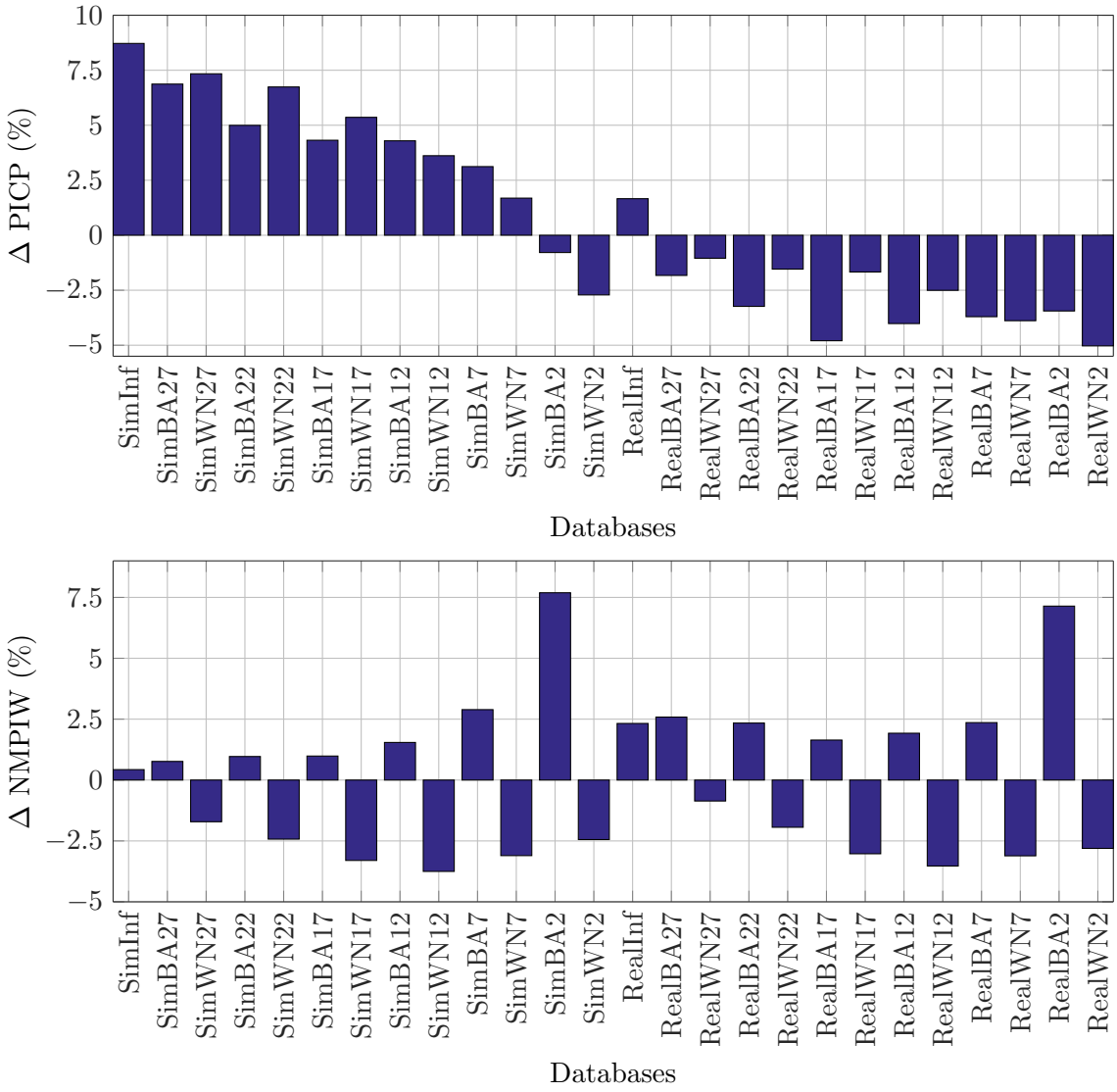

Figure 2.21: Difference between the PICP and NMPIW achieved in the different evaluation subsets and the PICP and NMPIW obtained for the development set using $= 5.6$ which provides a PICP=80% in the development set.

The correlation achieved with the conditional averaging for development set between $|C_{50,u} - \widehat{C_{50,u}}(y_u)|$ and CM_u is -0.95, which indicates that the confidence measure decreases as the estimation error increases, as it is shown in more detail in Fig. 2.23. The correlation differences between each evaluation subset and the development set are plotted in Fig. 2.24 in order to assess the repeatability of these results. On average, this difference is 0.39. However, there are 3 cases (SimWN27, SimWN22, and SimWN17) that show low positive correlation coefficients. This is due to the fact that there are few

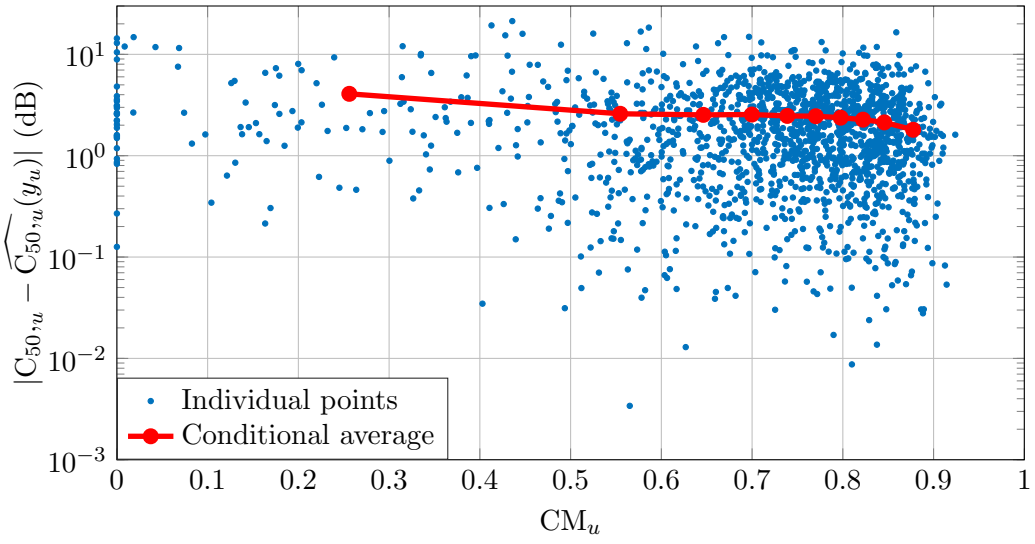

Figure 2.22: Confidence measures obtained in the development test set.

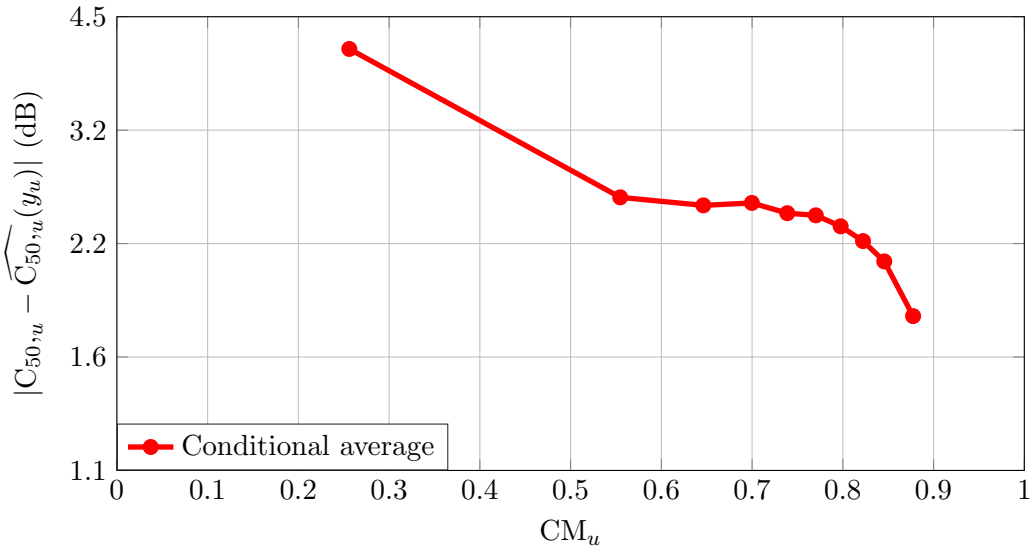

Figure 2.23: Zoom in the conditional averaging of the confidence measures obtained in the development test set.

low confidence points in these evaluation subsets, causing a noisy average for the lowest confidence measure band. In fact, the correlation coefficients are -0.59, -0.61 and -0.30 against the original coefficients 0.31, 0.40 and 0.18 when the averaged point corresponding to the lowest confidence measure band is ignored in SimWN27, SimWN22, and SimWN17 respectively.

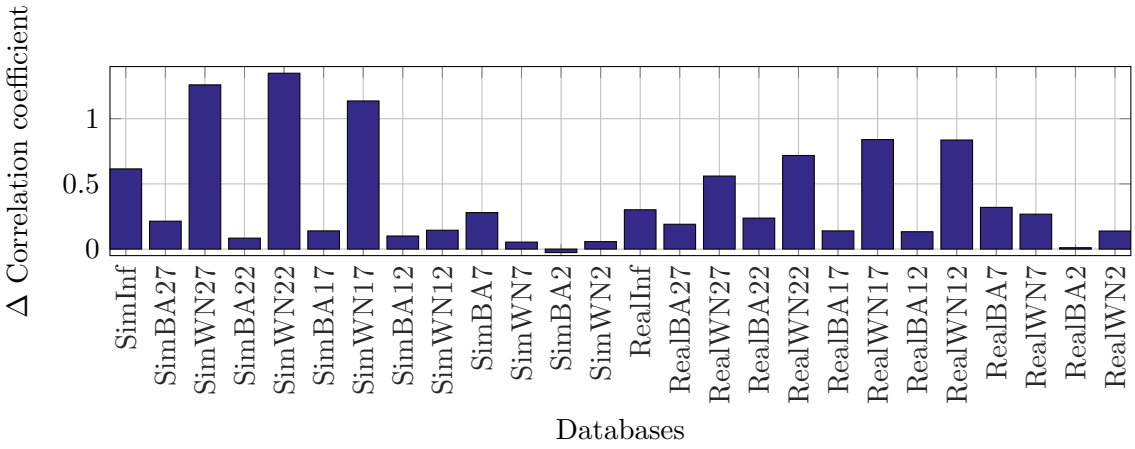

Figure 2.24: Difference between the correlation coefficients achieved in the individual evaluation subsets and those achieved in the development set. These correlation coefficients are obtained by conditional averaging the absolute estimation errors and the confidence measures obtained.

2.5.5 Conclusions

An approach to provide information about the accuracy of the C_{50} values obtained from NIRA has been proposed.

It has been shown that the prediction intervals, which provide an upper and lower bound of the estimate, can be derived from the standard deviation of the per frame estimations. On average, in 80% of the per utterance estimates, the ground truth is between the prediction intervals in the development set when computing these intervals as 5.6 times of the standard deviation. On average these intervals are placed ±4.69 dB from the estimation, which implies that the average width of the intervals is 30% of the ground truth range. Ideally, this width should be as close to 0% as possible, however the maximum recommended width depends on the application. Moreover, a confidence measure is derived from these prediction intervals which has shown high correlation coefficients with the C_{50} estimation error.

These methods were validated with an evaluation set comprised of 26 subsets that includes different RIRs, noise levels and noise types. The prediction intervals and confidence measures in this evaluation set using $\mathcal{K} = 5.6$ showed similar performance to the results obtained in the development set with the same value of $\mathcal{K}$, which suggests the

results, i.e. prediction intervals and confidence measure performance, are consistent over different databases and therefore repeatable.

The C_{50} estimator showed less accurate estimations for high levels of babble noise compared to the remainder set in the database, which was reflected in higher mean width of the prediction intervals.

2.6 NIRA DRR and T_{60} estimation

The use of the NIRA framework to estimate two additional room acoustic parameters is proposed: DRR and T_{60}. The approaches presented in this section are derived from the method that provides the lowest RMSD for C_{50} estimation, i.e. BLSTM. This technique is tested on the single-channel configuration of the ACE Challenge [21] organized by the IEEE Audio and Acoustic Signal Processing Technical Committee to compare directly different approaches to estimate DRR and T_{60} within the same framework.

2.6.1 Experimental setup

Unlike the definition of T_{60}, the definition of DRR is different than the one used in previous section, i.e. eq. (2.1). In this section the proposed DRR definition in the ACE Challenge is used:

$$\mathrm{DRR} = 10\log_{10}\left(\frac{\sum_{m=n_d-N_w}^{m=n_d+N_w} h^2(m)}{\sum_{m=0}^{m=n_d-N_w} h^2(m) + \sum_{m=n_d+N_w}^{\infty} h^2(m)}\right)\mathrm{dB}, \tag{2.22}$$

where N_w is the number of samples in a rectangular window of 8 ms and n_d is the time index (in samples) of the direct path arrival in the RIR $h(m)$. This time index n_d is determined by finding the maximum absolute value in the RIR.

2.6.1.1 Evaluation metrics

In addition to the box plots provided by the challenge [21] to compare the different approaches, the algorithms are compared in terms of RMSD. This metric is computed for

the DRR estimators as

$$\mathrm{RMSD}_{\mathrm{DRR}} = \sqrt{\frac{\sum_{u=1}^{N_{utt}} (\widehat{\mathrm{DRR}_{\mathrm{u}}} - \mathrm{DRR}_{\mathrm{u}})^2}{N_{utt}}} \text{ dB}, \tag{2.23}$$

where $\mathrm{DRR}_{\mathrm{u}}$ and $\widehat{\mathrm{DRR}}_{\mathrm{u}}$ are the ground truth and the estimated DRR respectively of the uth utterance and N_{utt} is the total number of utterances.

On the other hand, the RMSD of the T_{60} estimators is calculated as

$$\mathrm{RMSD}_{\mathrm{T60}} = \sqrt{\frac{\sum_{n=1}^{N_{utt}} (100 \cdot (\widehat{\mathrm{T}_{60,\mathrm{u}}} - \mathrm{T}_{60,\mathrm{u}})/\mathrm{T}_{60,\mathrm{u}})^2}{N_{utt}}} \text{ \%}, \tag{2.24}$$

where $\mathrm{T}_{60,\mathrm{u}}$ and $\widehat{\mathrm{T}_{60,\mathrm{u}}}$ are the ground truth and the estimated T_{60} respectively.

2.6.1.2 Data sets

The ACE Challenge database comprises 700 RIRs from 7 different rooms, ranging from small rooms such as offices to large spaces such as a lobby area. These responses are captured at 2 different positions by 5 different microphone configurations. In addition, anechoic speech from 5 male and 5 female talkers is provided as well as 3 types of noise (ambient, fan and babble). The ACE Challenge split this database in two parts: development set used to train or adapt the estimation method; and evaluation set employed only for evaluation purposes. The former set includes 2 rooms, 4 male speakers and the 3 types of noise at different SNR levels: 0 dB, 10 dB and 20 dB. The latter set comprises the remaining 5 rooms, 4 male and 4 female speakers and the 3 types of noise at different SNR levels: -1 dB, 12 dB and 18 dB. Although development and evaluation sets contain speech from the same speakers, the utterances in both sets are different. Figures 2.25 and 2.26 show the distribution of DRR and T_{60} targets respectively.

In order to learn models to estimate DRR and T_{60} for the single-channel full-band ACE Challenge task, all the development data from the different microphone configurations was split randomly into three parts: *trainSet*, *devSet* and *evalSet*. The *trainSet* comprises 70% of the files in the ACE Challenge development database, whereas *devSet* and *evalSet* comprise 20 % and 10 % respectively.

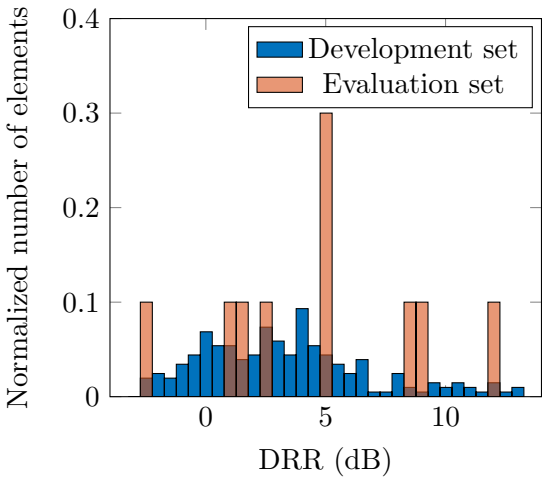

Figure 2.25: Distribution of the DRR targets in the ACE Challenge development and evaluation sets.

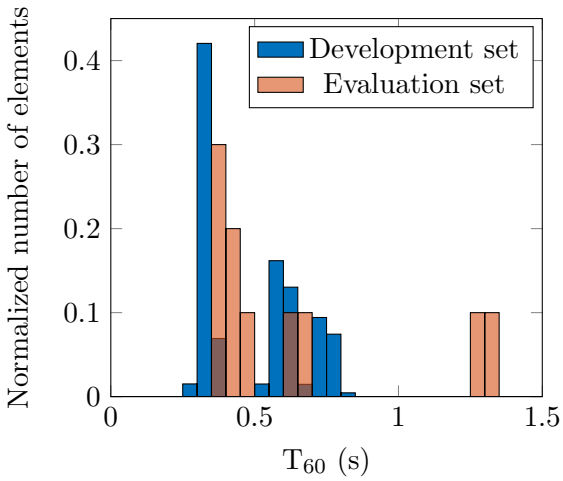

Figure 2.26: Distribution of the T60 targets in the ACE Challenge development and evaluation sets.

2.6.2 Method

The framework employed in this section is described in Section 2.3. This estimator was originally proposed for estimating C_{50} from 8 kHz speech signals. Therefore, an adaptation of the features has been developed here in order to process 16 kHz signals from the ACE Challenge. For speech signals sampled at 16 kHz, 10 LPCs and their corresponding LSFs are not sufficient to characterize the speech [86]. Therefore, for wide-band speech the order of the LPCs is increased to 20. Likewise, the number of Deep Scatter features used in NIRA is increased by a factor of $(\log_2(\mathcal{V}\cdot f_{s,wide-band})^2/2)/(\log_2(\mathcal{V}\cdot f_{s,narrow-band})^2/2)$ [47], where $\mathcal{V}$ is the frame size and $f_{s,wide-band}$ and $f_{s,narrow-band}$ are the wide-band and narrow-band sampling frequencies respectively. The total number of DSS features increases then from 21 to 28. Hence the features per frame comprises 134 elements.

Since ACE Challenge data assumes that the room acoustic properties remain unchanged within each utterance, only the temporal average for each utterance of all per frame estimations is considered.

Different architectures of the BLSTM are explored with one to four layers including 64, 128 and 256 neurons per layer and a minibatch size of 25, 50, 100 and 200 samples.

The minibatch size refers to the number of samples, in this case the number of utterances, employed in the stochastic gradient descent to update the network weights in each iteration. The usage of minibatches in the speeds up the training time. Three different configurations were explored using this framework which are described in the following subsections.

2.6.2.1 NIRAv1

This configuration is based on training the NIRA framework presented in Fig. 2.8 using only the ACE Challenge development database. In this case, *trainSet* is used to train the model and *devSet* is employed to validate the model, then the selected model is the one that minimizes the estimation error in *devSet*.

2.6.2.2 NIRAv2

This configuration employs the NIRA framework shown in Fig. 2.8 trained on three different databases in order to introduce new data in the model which could generalize the model to a wider range of scenarios. In this case 60% of the files are extracted from the ACE Challenge development database, 20% of the files from the REVERB Challenge database and the remainder of the files are taken from a database created with the TIMIT database [71] and real impulse responses from MARDY [3], SMARD [76], C4DM RIR [75] and REVERB Challenge [74] database presented in Section 2.4.1.2.3. Similarly, *devSet* is created with the same proportions and from the same databases but the total number of files is 30% of *trainSet*.

2.6.2.3 NIRAv3

This configuration follows the structure shown in Fig. 2.27. It is based on training 4 different BLSTM models using different data: NIRAv1; NIRA_{α} using the whole REVERB Challenge development set; NIRA_{β} and NIRA_{γ} employing real and simulated RIRs respectively from the evaluation set presented in Section 2.4.1.2.3. These 4 estimators are combined by averaging the per frame estimations of each utterance and by training a SVR

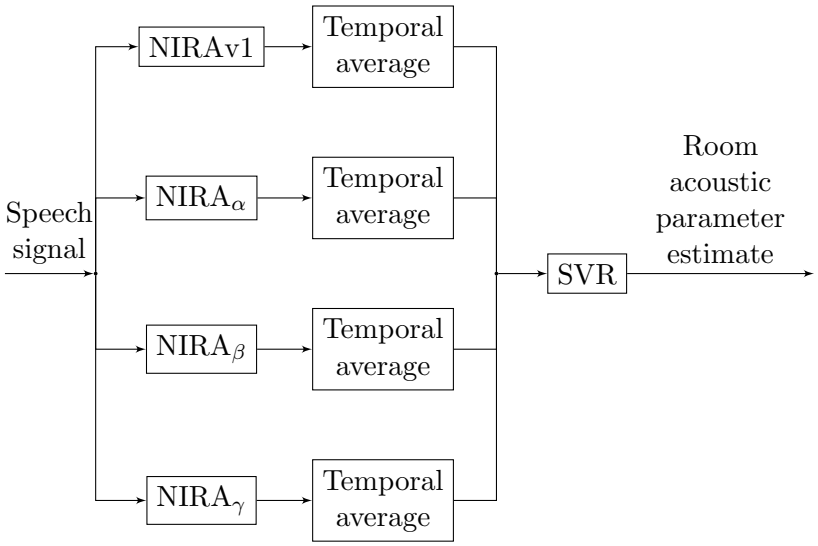

Figure 2.27: The NIRAv3 configuration for DRR and T_{60} estimation.

model [87] with the 4-dimensional estimate vector obtained from the individual estimators. The training data for this SVR is *devSet* from NIRAv1 and *evalSet* is used for validation purposes.

2.6.3 Performance evaluation

The evaluation results for the different approaches are shown in this section. These approaches are tested on two datasets: *evalSet* described in Section 2.6.2.1 and the ACE Challenge evaluation set.

Several topologies are trained for each model and the topology that provides the lowest estimation error on *devSet* is selected. The selected topologies for each model are displayed in Table 2.10.

2.6.3.1 Performance in *evalSet*

Table 2.11 shows the performance of the three approaches in terms of RMSD on the *evalSet* dataset introduced in Section 2.6.2.1. NIRAv1 and NIRAv3 show the best performance for DRR estimation and NIRAv2 the highest estimation error deviation. Figure 2.28 displays the box plot for the same dataset. NIRAv2 shows a wider Interquartile range (IQR)

Model	Target	Number layer	Number neurons	Minibatch size
NIRAv1	T_{60}	2	128	25
	DRR	3	64	25
NIRAv2	T_{60}	4	64	25
	DRR	3	64	25
$NIRA_{\alpha}$	T_{60}	4	64	100
	DRR	4	64	100
$NIRA_{\beta}$	T_{60}	4	64	200
	DRR	4	64	50
$NIRA_{\gamma}$	T_{60}	4	64	200
	DRR	4	64	25

Table 2.10: Topologies for each trained model.

and a negative bias which explains the higher RMSD value compared to the other two configurations. Regarding T_{60} estimation, Table 2.11 indicates that the best approach is NIRAv1, whereas NIRAv3 provides the lowest performance mainly due to the bias and the wide IQR displayed in Fig. 2.29.

Configuration	$RMSD_{DRR}$ (dB)	$RMSD_{T_{60}}$ (%)
NIRAv1	0.64	3.18
NIRAv2	0.92	3.66
NIRAv3	0.63	7.15

Table 2.11: RMSD of the three approaches to estimate DRR and T_{60} using *evalSet* dataset.

In order to find statistical differences in the estimation errors, the Wilcoxon matched pair signed-rank test [88] is applied. This approach is an optimal test to compare paired observations when normality of these observations can not be guaranteed [89]. This hypothesis testing method is a non-parametric approach to test the null hypothesis that the pair differences are symmetrically distributed with respect to the median equal to 0. The p-values obtained for each of the performed tests, shown in Table 2.12, are multiplied by the number of pair-wise comparison, i.e. 3, following the Bonferroni correction for multiple comparison [90]. These results indicate that the null hypothesis can be rejected in all the cases at a significance level of 0.05, hence the performance of each configuration is statistically different.

The quality of an estimator can be often best understood as the estimation variance

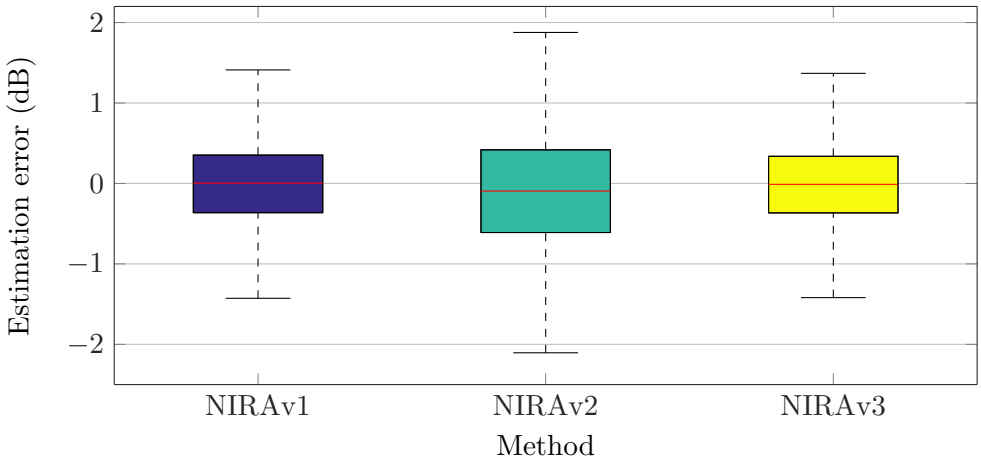

Figure 2.28: Distribution of the DRR estimation errors for each configuration using *evalSet*. The edges of the boxes indicate the lower and upper quartile range, while the horizontal lines inside the boxes represent the medians for each configuration. Moreover, the horizontal lines outside the boxes indicate the estimation error up to 1.5 times the interquartile range.

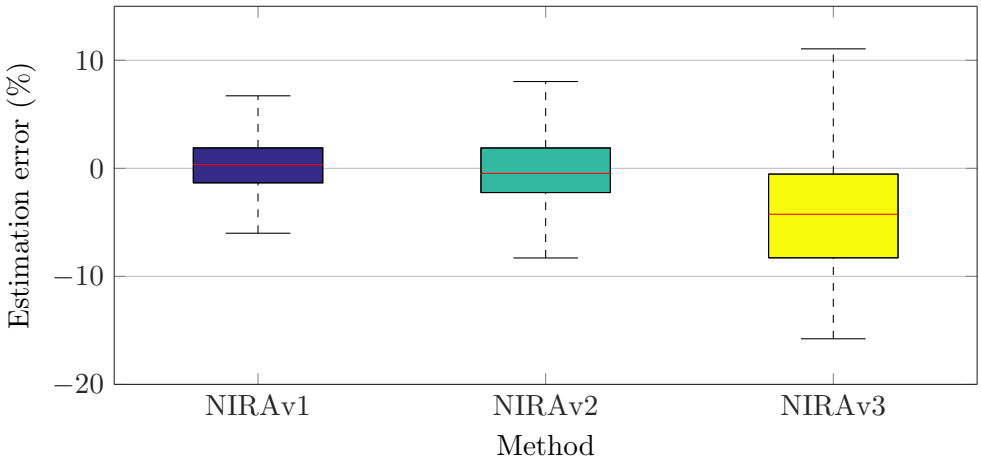

Figure 2.29: Distribution of the T_{60} estimation errors for each configuration using *evalSet*.

[91]. The *evalSet* would be used in order to compensate for possible bias, however Fig. 2.28 and Fig. 2.29 show that the bias is negligible in the cases of interest, i.e. low variance. Consequently, no compensation of bias is applied.

2.6.3.2 Performance in ACE Challenge Evaluation set

Table 2.13 shows the performance of the three approaches on the ACE Challenge evaluation dataset. NIRAv3 and NIRAv1 still provide the best performance when estimating

Set 1	Set 2	Estimated parameter	*p*-value
NIRAv1	NIRAv2	DRR	<0.001
NIRAv1	NIRAv3	DRR	<0.001
NIRAv2	NIRAv3	DRR	<0.001
NIRAv1	NIRAv2	T_{60}	<0.001
NIRAv1	NIRAv3	T_{60}	<0.001
NIRAv2	NIRAv3	T_{60}	<0.001

Table 2.12: *p*-values obtained with the Wilcoxon matched pair signed-rank tests and applying Bonferroni correction where the sets represent the approaches employed to compute the estimation errors on the *evalSet* dataset.

DRR and T_{60} respectively on this dataset, however the deviations are considerably increased. This can be due to an overfitting problem since *devSet* and *trainSet* contain similar utterances obtained from same RIRs.

Configuration	$\mathbf{RMSD_{DRR}}$ (dB)	$\mathbf{RMSD_{T_{60}}}$ (%)
NIRAv1	3.87	43.19
NIRAv2	3.85	44.80
NIRAv3	3.84	44.18

Table 2.13: RMSD of the three approaches to estimate DRR and T_{60} using ACE Challenge evaluation set.

Figure 2.30 shows the distribution of the DRR estimation error for each configuration. The three configurations present similar distributions, however NIRAv3 is less biased which is in accordance with the results displayed in Table 2.13. Figure 2.31 shows the box plot for each configuration proposed to estimate T_{60}. NIRAv3 presents the higher interquartile range and NIRAv1 the least biased estimation, which is reflected in the deviation shown in Table 2.13. In general, the box plots show that T_{60} is almost always underestimated which is due to the lack of RIRs with high T_{60} in the ACE Challenge development set as illustrated in Fig. 2.26. This issue leads to underestimating the utterances with high T_{60} included in the ACE Challenge evaluation set, thus obtaining a significant negative bias.

A hypothesis testing is performed, as in previous section, using the Wilcoxon matched pair signed-rank test to find statistical differences amongst the different performances. The *p*-values obtained on these tests, shown in Table 2.14 after applying Bonferroni correction, indicate that all the performances are statistically different at sig-

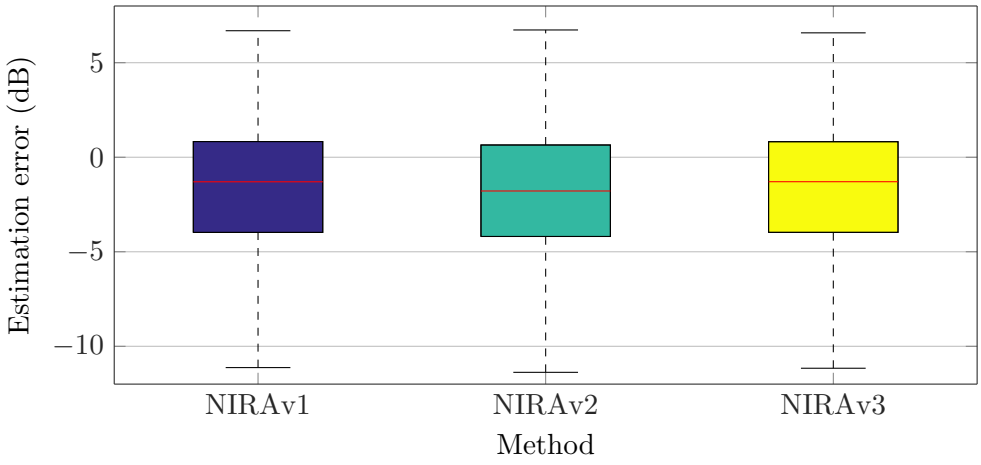

Figure 2.30: Distribution of the DRR estimation errors for each configuration using ACE Challenge evaluation dataset.

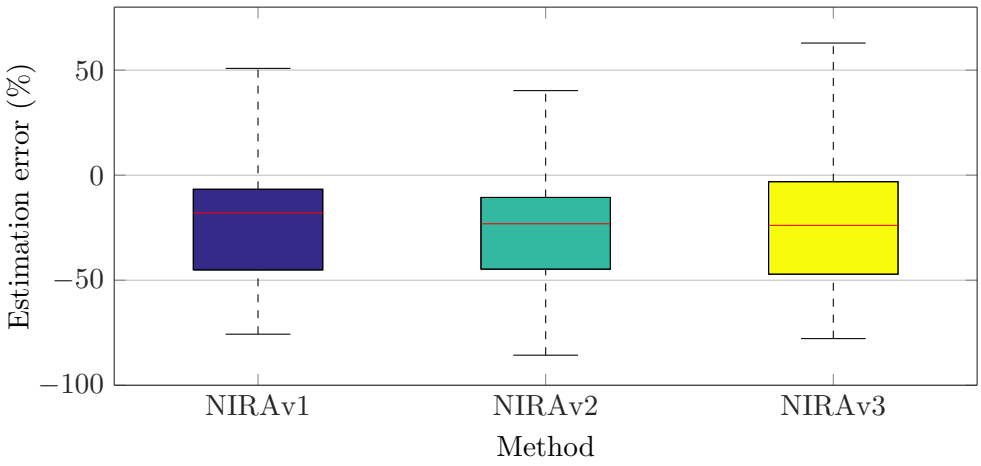

Figure 2.31: Distribution of the T_{60} estimation errors for each configuration using ACE Challenge evaluation dataset.

nificance level of 0.05 except for the performances of NIRAv1 and NIRAv3 when estimating DRR.

An analysis of the performance of the best approaches to estimate DRR and T_{60} is shown in Fig. 2.30 and 2.31 respectively for each noise condition. These figures suggest that babble noise provides the lowest RMSD for DRR estimation whereas fan noise in the recordings brings higher DRR estimation errors. On the contrary, fan noise provides the lowest T_{60} deviation and babble noise brings the highest T_{60} estimation errors.

An improvement when estimating T_{60} is achieved by modifying the default cost

Set 1	Set 2	Estimated parameter	*p*-value
NIRAv1	NIRAv2	DRR	<0.001
NIRAv1	NIRAv3	DRR	2.46
NIRAv2	NIRAv3	DRR	<0.001
NIRAv1	NIRAv2	T_{60}	<0.001
NIRAv1	NIRAv3	T_{60}	<0.001
NIRAv2	NIRAv3	T_{60}	<0.001

Table 2.14: *p*-values obtained with the Wilcoxon matched pair signed-rank tests and applying Bonferroni correction where the sets represent the approaches employed to compute the estimation errors on the ACE Challenge evaluation dataset.

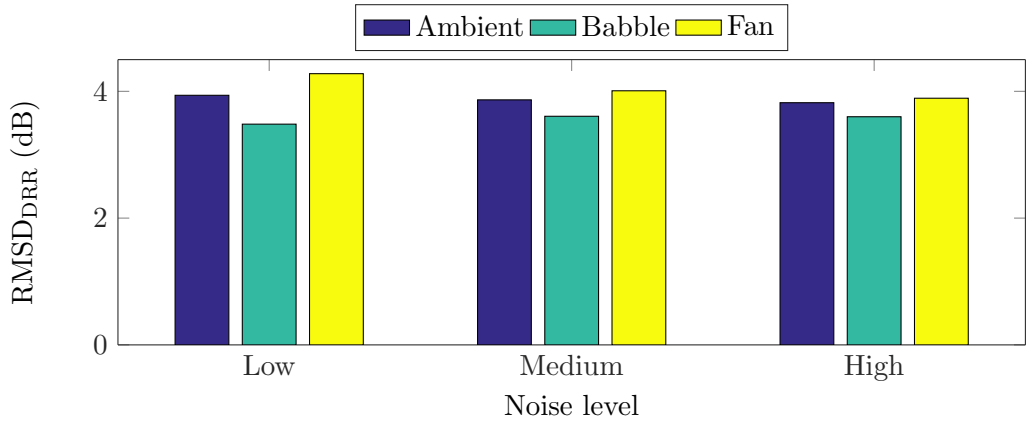

Figure 2.32: Performance of NIRAv3 estimating DRR on the ACE Challenge evaluation dataset for different noise conditions.

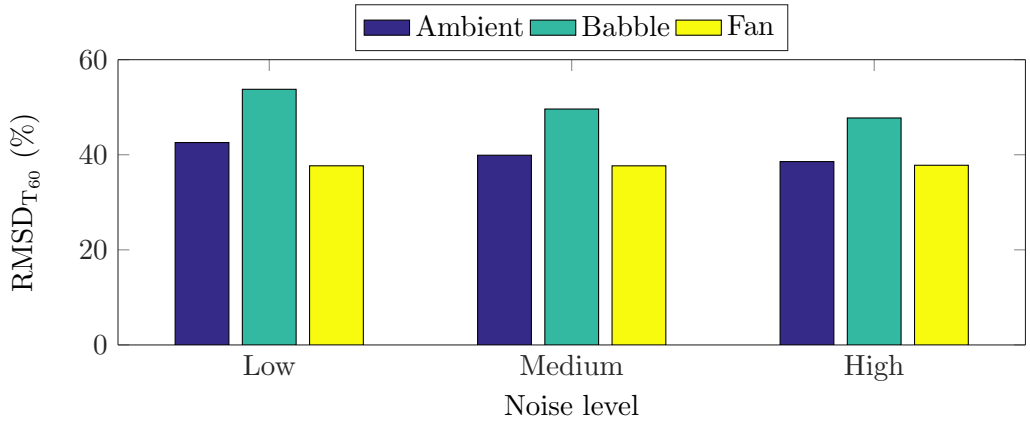

Figure 2.33: Performance of NIRAv1 estimating T_{60} on the ACE Challenge evaluation dataset for different noise conditions.

function of the BLSTM toolkit[3]. The default cost function in this toolkit is the Sum of

[3]http://sourceforge.net/projects/currennt/

Squared Errors (SSE) which is defined as follows

$$\mathrm{SSE} = \sum_{u=1}^{N_{utt}} (\widehat{\mathrm{T}_{60,\mathrm{u}}} - \mathrm{T}_{60,\mathrm{u}})^2. \tag{2.25}$$

However, the evaluation metric to measure the performance of the T_{60} estimator, (2.24), is based on the percentage error rather than the absolute error in order to penalize large errors when estimating low T_{60} ground truth values. Thus, the SSE is substituted by the Sum of Squared Percentage Errors (SSPE)

$$\mathrm{SSPE} = \sum_{u=1}^{N_{utt}} \left((\widehat{\mathrm{T}_{60,\mathrm{u}}} - \mathrm{T}_{60_\mathrm{u}})/\mathrm{T}_{60,\mathrm{u}} \right)^2. \tag{2.26}$$

Table 2.15 shows the performance achieved estimating T_{60} with the same BLSTM topology employed for NIRAv1 for two different cost functions: SSE and SSPE. The SSPE cost function reduces by approximately 1% the $\mathrm{RMSD}_{\mathrm{T}_{60}}$ and the estimation error bias while it increases the correlation ρ of the estimation with the ground truth.

Cost function	**$\mathrm{RMSD}_{\mathbf{T_{60}}}$ (%)**	ρ	Bias (%)
SSE	43.19%	0.26	14.54
SSPE	42.11%	0.30	13.25

Table 2.15: Performance comparison of different cost functions employed in training to estimate T_{60}.

An alternative set of features based only on the absolute values of the spectral bins extracted from the STFT with frame window of 20 ms is used to train a BLSTM with the same topology as NIRAv1. In this case, $\mathrm{RMSD}_{\mathrm{DRR}} = 4.78$ dB which indicates that the set of features presented in Table 2.5 is more suitable for this task.

2.6.4 Conclusions

Three data-driven approaches have been presented to estimate full-band DRR and T_{60} from single-channel reverberant speech. The first two approaches are based on training a BLSTM with two different datasets. Additionally, the third approach is based on combining BLSTMs trained with different datasets by employing a SVR. The best DRR estima-

tion performance was achieved with NIRAv3, $\text{RMSD}_{\text{DRR}} = 3.84$ dB with IQR = 4.79 dB and median of -1.3 dB. This best result is based on training with different databases several BLSTMs and combining their individual time averaged estimations with a SVR. On the other hand, NIRAv1 provides the best T_{60} estimation performance, $\text{RMSD}_{T_{60}} = 43.19$ % with IQR = 44 % and median of -23.88 %. This configuration is based on training a BLSTM employing only the ACE Challenge development dataset.

Moreover, the performance of these approaches was tested with 10 % of the ACE Challenge development files, not previously used in the training process, i.e. *evalSet*. The best performance of DRR and T_{60} was obtained with NIRAv3 and NIRAv1 respectively, as it occurs on ACE Challenge evaluation dataset. However, the deviations were considerably lower, $\text{RMSD}_{\text{DRR}} = 0.63$ dB with IQR = 0.7 dB and median of -0.01 dB for DRR estimation and $\text{RMSD}_{T_{60}} = 3.18$ % with IQR = 3.23 % and median of 0.34 % for T_{60} estimation. This suggests that the trained models are overfitting the ACE Challenge development dataset and they are not totally able to generalize the estimation problem to a complete new dataset, i.e. the ACE Challenge evaluation dataset. This overfitting could potentially be reduced by including female talkers in *trainSet* and *devSet* or including new RIRs in *devSet*.

The NIRAv3 system was found to provide the best performance in the single-microphone full-band DRR estimation task and second best performance among all the submitted methods to the ACE Challenge which include multi-microphones approaches. However, in the single-microphone full-band T_{60} task, NIRAv1 was found in the context of ACE Challenge to be in the bottom quartile for performance [92].

Chapter 3

Reverberant speech recognition using spatial features

In this chapter, the impact of reverberation on phoneme recognition is analysed for numerous reverberant conditions, and a metric to estimate the confusability of each phoneme depending on the reverberation level is derived. This metric is then employed to improve ASR performance and finally an acoustic model switching method based on C_{50} estimation is introduced to recognize reverberant speech.

The research presented in this chapter relates in part to the following publications [13, 9, 15].

3.1 Introduction

ASR is increasingly being used as a tool for a wide range of applications in diverse acoustic conditions (e.g. health care transcriptions, automatic translation, voicemail-to-text, voice interface for command and control, etc.). Of particular importance is distant speech recognition, where the user can interact with a device placed at some distance from the user. Distant speech recognition is essential for natural and comfortable human-machine voice interfaces such as used in, for example, the automotive sector and smart-phone applications. As a result of this interest, multiple challenges have been launched in the

research community in recent years such as REVERB Challenge[1] or CHiME3 Challenge[2] to promote new approaches to robustly process noisy reverberant data.

3.1.1 Technical background

ASR attempts to convert speech into its corresponding transcription. First attempts to carry out this task can be found back on the fifties when researchers at *Bell Labs* built a recognizer based on phonetic elements. The goal of this system was to recognize isolated digits based on the formant frequencies [93]. Later on this decade, statistical information with regards to phoneme combination were added thus allowing isolated word recognition. From this point onwards, different techniques (e.g. Dynamic Time Warping, Linear Predictive Coding, Hidden Markov Models, Language Models, Deep Neural Networks) were incorporated to speech recognizers allowing nowdays to perform Large Vocabulary Continuous Speech Recognition (LVCSR).

Figure 3.1 shows a generalized ASR block diagram. These systems can be usually split in two main blocks [94]: front-end and back-end. The former pre-processes the input signal and extracts a feature vector $\mathbf{O}$ for each analysed speech frame which contains relevant information to discriminate the different acoustic units, e.g. phonemes, triphones. The latter comprises a set of information resources which model the statistical distribution of the feature vector for each acoustic unit (Acoustic Model), represent the words as a concatenation of acoustic units (Dictionary) and indicate which sequence of words $\boldsymbol{\mathcal{W}}$ are likely to occur (Language Model). This information is pre-trained from large databases and tailored to the application where it is used.

The ASR goal is to select the word sequence $\boldsymbol{\mathcal{W}} = \{w_1, w_2, \cdots w_{N_{wrd}}\}$ so as to maximize the posterior probability of $\boldsymbol{\mathcal{W}}$ given the observations $\mathbf{O} = \{\mathbf{o}_1, \mathbf{o}_2, \cdots \mathbf{o}_{N_{frm}}\}$. Therefore the chosen word sequence is given by the solution of

$$\arg\max_{\boldsymbol{\mathcal{W}}} \frac{p(\boldsymbol{O}|\boldsymbol{\mathcal{W}})p(\boldsymbol{\mathcal{W}})}{p(\boldsymbol{O})}. \tag{3.1}$$

[1] http://reverb2014.dereverberation.com/
[2] http://spandh.dcs.shef.ac.uk/chime_challenge/

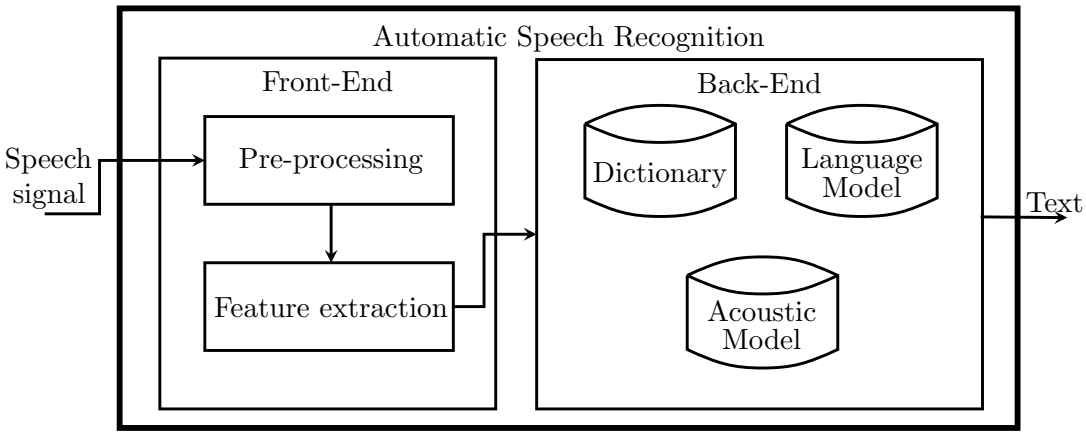

Figure 3.1: Speech recognition diagram.

Since $\mathbf{O}$ in (3.1) is fixed in the maximization problem, expression (3.1) is equivalent to

$$\arg\max_{\mathbf{W}} p(\mathbf{O}|\mathbf{W})p(\mathbf{W}), \tag{3.2}$$

where the likelihood $p(\mathbf{O}|\boldsymbol{\mathcal{W}})$ is extracted from the acoustic model and the prior $p(\boldsymbol{\mathcal{W}})$ is retrieved from the language model.

The design of acoustic and language models, from which (3.2) can be solved, is of critical importance in ASR. The acoustic model should take into consideration speech variations, such as speakers' pronunciation, and environmental variations, such as noise or reverberation. Likewise, the language model should take into account the topic characteristics, such as the words in the vocabulary. Therefore, large databases are created including all these considerations to build these models. In addition to this training process, dynamic model adaptation is also critical to improve the ASR performance dynamically as it is being used.

The acoustic model is commonly based on a combination of HMM-GMM [45] or HMM-Deep Neural Network (DNN) [95] where the likelihood of the acoustic units is computed from GMM or DNN and the transition probabilities between these units are modelled with HMM. Assuming conditional independence of the feature vectors and that

a first-order Markov chain is employed, the acoustic probability can be computed as [6]

$$p(\mathbf{O}|\mathbf{W}) = \sum_{\mathcal{S}} p(\mathbf{O}|\mathcal{S},\mathbf{W})p(\mathcal{S}|\mathbf{W}) \tag{3.3}$$

$$= \sum_{\mathcal{S}} \pi(\mathcal{S}_0) \prod_{l=1}^{N_{frm}} a_{\mathcal{S}_{l-1}\mathcal{S}_l} \prod_{l=0}^{N_{frm}} p(\mathcal{S}_l|\mathbf{o}_l) \tag{3.4}$$

$$\tag{3.5}$$

where $\mathcal{S}$ represents a HMM state sequence, $\pi(\mathcal{S}_0)$ is the initial state probability, $a_{\mathcal{S}_{l-1}\mathcal{S}_l}$ is the transition probability from state $\mathcal{S}_{l-1}$ to state $\mathcal{S}_l$ and $p(\mathcal{S}_l|\boldsymbol{o}_l)$ is the observation probability computed from a model (e.g. GMM) at frame l.

Language models are usually created in a probabilistic framework using n-grams which provide the probability of the next word given the $n-1$ previous words

$$p(w_n) = p(w_n|w_1, w_2, \cdots, w_{n-1}) \tag{3.6}$$

where n, in this special case of n-grams context, is the notation employed to indicate the number of previous words considered in the computation of the language probability. More recently language models are built with RNN [66] where the context is not limited to n. In limited domain applications such as question answering tasks, simpler approaches as weighted finite-state grammars [96] can be used.

Solving (3.2) requires evaluating all possible combinations of word sequences to find the sequence with the highest a posteriori probability. In LVCSR applications, where the number of words is above 20000 words [96], this exhaustive search, also known as decoding, becomes impracticable. Instead, different search algorithms [94], such as Viterbi algorithm, can be applied in order to reduce the possible combinations and thus decreasing the decoding computational cost.

As introduced in Section 2.1.1, reverberant speech is created in confined spaces by multipath sound propagation from source to receiver which creates multiple delayed and attenuated replicas of the original sound [97]. This convolutional noise significantly decreases ASR performance in distant-talking scenarios [6] [98]. The performance degra-

dation is mainly due to the mismatch between the ASR acoustic model and the reverberant signal. Unlike stationary noise, reverberation creates highly non-stationary and correlated noise due to the temporal smearing of the signal and thus specific methods need to be applied in order to overcome the performance degradation of reverberant speech in ASR.

3.1.2 Literature review

ASR techniques robust to reverberation can be divided in two main groups [98, 99, 100]: front-end-based and back-end-based. The former approach suppresses the reverberation in the feature domain, therefore the processing is performed after the feature extraction. Li et al. [101] propose to train a sparse transformation to estimate the clean feature vector from the reverberant feature vector based on learning jointly a sparse transformation of clean and reverberant features. In [102] a model of the noise is estimated from observed data by considering the late reverberation as additive noise and then the feature vector is enhanced by applying vector Taylor series. A feature transformation based on discriminative training criterion inspired by Maximum Mutual Information is suggested in [103]. Additional features related to the amount of diffuse noise in each frequency bin and frame are employed in [104] to improve deep neural network based ASR accuracy in noisy and reverberant environments. Yoshioka and Gales [105] present several front-end approaches such as feature transformation or feature set expansion that are tailored to deep neural network acoustic models employed for distant-talking recognition.

The latter approach, back-end-based, modifies the acoustic models or the observation probability estimate to suppress the reverberation effect. Sehr et al. [106] suggest to adapt the output probability density function of the clean speech acoustic model to the reverberant condition in the decoding stage. A selection of different acoustic models trained for specific reverberant conditions using an estimation of T_{60} is proposed in [107]. In [31] several RIRs are built employing a generalised statistical RIR, derived from the generalised Schroeder's model, which includes two tuning parameters: early reflection period and late reverberation attenuation. These RIRs are used to create reverberant acoustic models which are then selected in the recognition phase using ground truth T_{60}. The tuning pa-

rameters are selected such that they provide the highest recognition rate on a reverberant test set created with measured RIRs. The early-to-late reverberation ratio, considering the first 110 ms of the RIR as part of the early reverberation, is used in [108] instead of T_{60} to select between different reverberant acoustic models. In [109] the likelihood scores of the ASR acoustic models based on GMM are maximized to select the optimum acoustic model. An adaptation of multiple reverberant acoustic models trained with different T_{60} values is proposed in [110]. The mean vector of the optimal adapted model is estimated in a maximum-likelihood sense from the reverberant models. Estimates of T_{60} and DRR are employed in [37] to estimate the type of room that created the reverberant signal and use the acoustic model created with the estimated room type for recognition. The idea in [32] is to add to the current state the contribution of previous acoustic model states using a piece-wise energy decay curve which considers the early reflections and late reverberation as different contributions.

In addition to front-end-based and back-end-based approaches, signal-based methods are intended to dereverberate the acoustic signal in the time domain, before being processed by the ASR feature extraction module [22]. In [111] a complementary Wiener filter is proposed to compute suitable spectral gains which are then applied to the reverberant signal to suppress late reverberation. In [112] a denoising autoencoder is used to clean a window of spectral frames and then overlapping frames are averaged and transformed to the feature space. Based on the fact that linear prediction residuals of voiced speech contain strong peaks, adaptive filters can be used to suppress the reverberation in the signal by maximizing the kurtosis of the linear prediction residual of the processed signal [113]. In [114] a non-linear transformation is modelled with DNN to map noisy reverberant speech coefficient, i.e. log-magnitude spectrum, to clean speech coefficient. All these three approaches may be combined to create complex robust systems [115] [65] [116].

Additionally, ASR techniques robust to reverberation can be also classified according to the number of microphones used to capture the signal such as single-channel methods [102] [117] [112] [118] or multi-channel techniques [101] [115] [119] [120].

3.2 Phoneme analysis of reverberant speech recognition

Phoneme intelligibility degradation for humans due to reverberation was investigated in [4]. The authors showed how reverberation degrades human intelligibility and that resulting errors have the same distribution compared to non-reverberant environments. ASR performance also degrades in the presence of reverberation although the behaviour compared to human intelligibility is different: the indicative error rate is higher in ASR compared to human listeners [121] [122]. In [29] the performance of a digit recognizer is analysed for different reverberation levels obtained by carefully modifying the RIR. The authors demonstrated that the first 50 ms of the RIR barely affect ASR performance whereas the remainder of the RIR has a significant detrimental impact. Tsilfidis et al. [27] investigated the reverberation impact on phoneme recognition showing the performance achieved for the different reverberation levels considered.

The impact of reverberation on phoneme recognition is analysed in this section for numerous reverberant conditions, with a special focus on the confusion found between phonemes. This analysis provides insights into the ASR robustness of each phoneme for different reverberation levels. Furthermore, a model to estimate the confusability of each phoneme depending on the reverberation level is derived from an analysis of the confusion matrices.

3.2.1 Experimental setup

The TIMIT database [71] is used in all the experiments performed in this section. This database is phonetically tagged and it contains a good phonetic coverage of American English [123] providing a rich contextual phoneme diversity [124]. These characteristics provide an ideal framework to analyse the reverberation impact per phoneme since each of these phonemes appears in many different contexts.

Two different speech recognizers are implemented to analyse the effect of reverberation in phoneme recognition. First, a Context-Independent GMM-HMM phone recognizer based on HTK (CI-HTK) [45] is trained following the recipe suggested in [27]. Second,

an alternative Context-Independent GMM-HMM phone recognizer based on Kaldi toolkit (CI-KALDI) and Context-Dependent GMM-HMM phone recognizer based on Kaldi toolkit (CD-KALDI) are built using Kaldi recipe s5 for TIMIT [42]. In all cases, a single-pass decoding without lattice re-scoring or feature transformation is performed in order to reduce the computational cost. The motivation for using a phoneme recognition in this analysis is to avoid potential impact of language model or dictionary rules in the recognition performance and more specifically analyse the impact of acoustic distortions.

The whole TIMIT test set, excluding the 2 dialect sentences (SA), is divided into two independent sets: the Non-reverberant development set (ClnDev) and the Non-reverberant evaluation set (ClnEval). The latter comprises the TIMIT core test (192 utterances) and the former includes the remaining test recordings (1152 utterances). The initial 61 phonemes in both test sets are collapsed into a set of 39 phonemes [125]. ClnDev is convolved with 140 simulated RIRs, which are uniformly distributed with C_{50} values from -3 dB to 40 dB as shown in [12], to create the Reverberant development set (RevDev). The Reverberant evaluation set (RevEval) is generated by convolving ClnEval with 28 simulated RIRs spanning the C_{50} interval [-3dB, 40 dB] and with all real impulse responses (72 RIRs) from MARDY database [3]. The resulting reverberant sets, RevDev and RevEval, are approximately 138 hours and 16 hours long respectively, which cover a wide range of reverberant scenarios.

The parameter used to measure the reverberation level is C_{50} (2.3) as it has been shown in Section 2.2.4.1 to be highly correlated with ASR performance [12] [27].

3.2.2 Impact of reverberation on ASR performance

In this section the performance of phoneme recognition is shown for a broad range of reverberation levels as well as the phoneme misclassification for clean and reverberant environments. The ASR performance is computed following (2.6).

The PER achieved with ClnDev and RevDev for different ASR configurations is displayed in Table 3.1, which shows a clear ASR performance reduction due to the presence of reverberation. Figure 3.2 describes in more detail the relative Phoneme Error Rate

degradation (rΔPER) obtained for different reverberation levels following

$$\text{r}\Delta\text{PER}(\%) = \frac{\text{PER}_{\text{RevDev}} - \text{PER}_{\text{ClnDev}}}{100 - \text{PER}_{\text{ClnDev}}} \cdot 100. \tag{3.7}$$

In the case of low reverberation levels (i.e. $C_{50}\approx$40 dB) the performance of the different phoneme recognizers is scarcely affected. However, an increment of the reverberation level clearly leads to a significant degradation which shows the importance of understanding the reverberation impact on ASR.

	CI-HTK	**CI-KALDI**	**CD-KALDI**
ClnDev	40.2%	35.52%	33.59%
RevDev	66.8%	62.28%	59.45%

Table 3.1: Phoneme error rate achieved with ClnDev and RevDev.

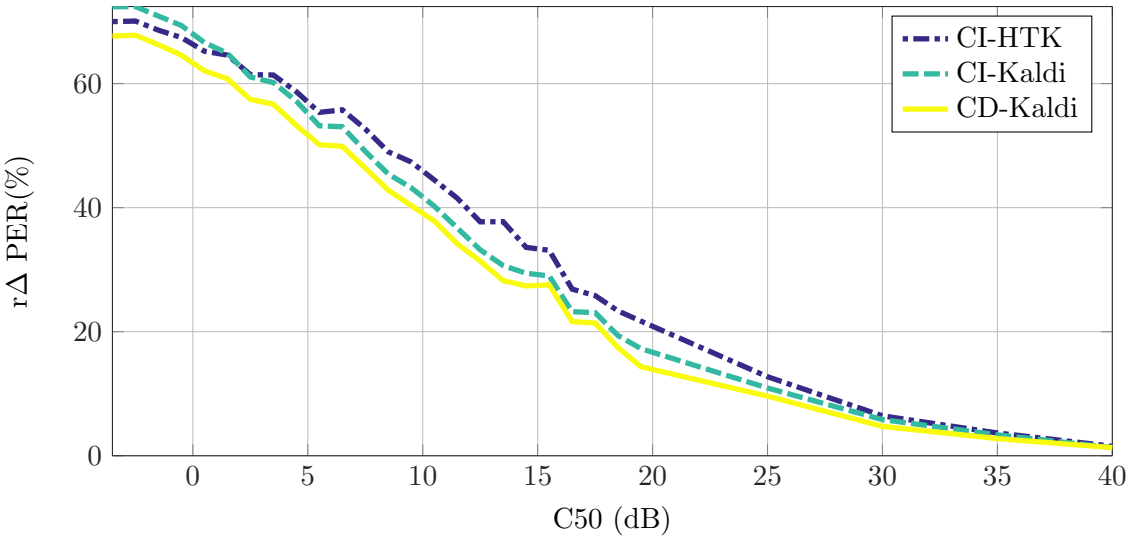

Figure 3.2: Relative phoneme error rate degradation rΔPER vs. reverberation level C_{50}.

The performance of each phoneme in reverberant environments is presented in Fig. 3.3 and Fig. 3.4 which plot the confusion matrix obtained with ClnDev and RevDev respectively. These matrices are obtained with the ASR system that provides the best performance in these experiments: CD-KALDI. The matrices are normalized horizontally and consequently each cell, for instance row k starting from top and column $\tilde{k}$ starting from left, represents the likelihood of recognizing the phoneme $R_{\tilde{k}}$ given the true phoneme

T_k. As a result, the main diagonals in the matrices represent the likelihood of correctly recognizing the given phoneme that is $P(R_k|T_k)$ where k is the phoneme index. The /sil/ label represents a pause. In addition to the 39 phonemes a new label /blk/ representing a blank is included in the matrices to take into account the deletions and insertions. Therefore the last row represents the insertions and the last column the deletions.

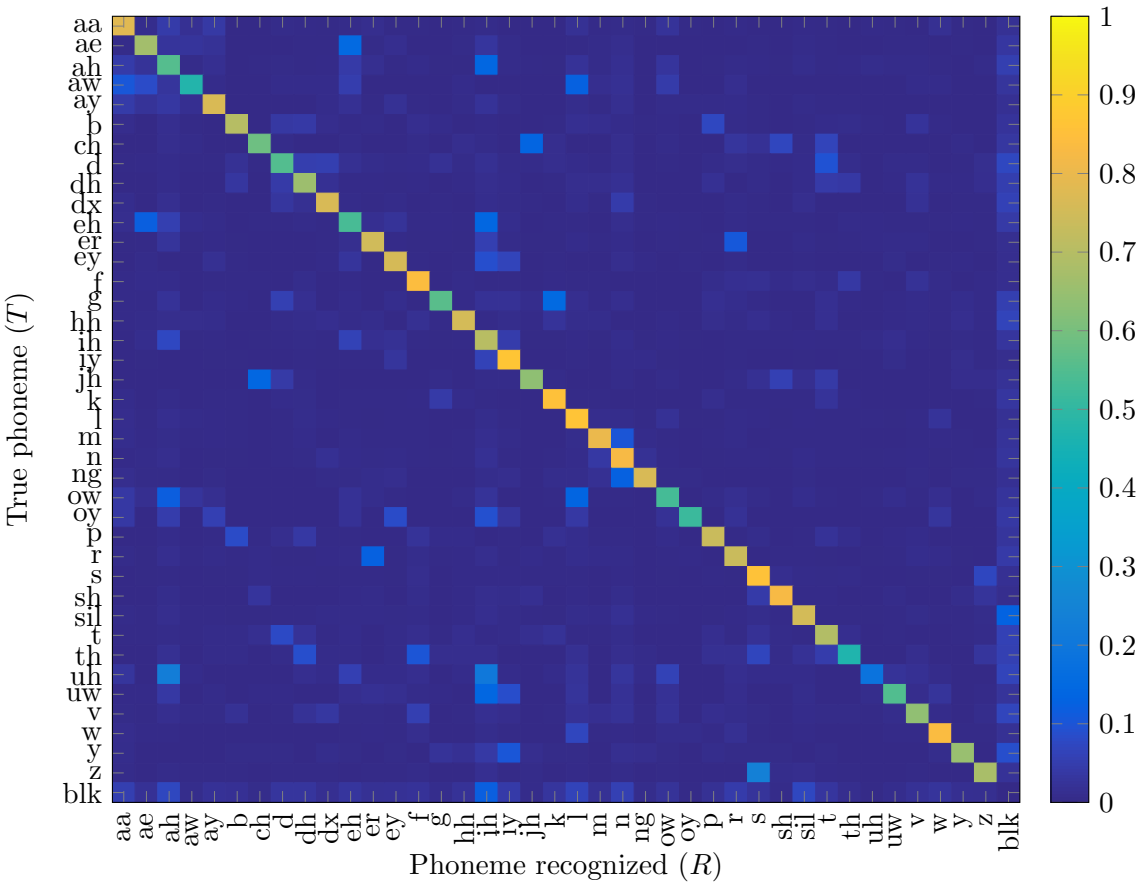

Figure 3.3: Phoneme confusion matrix obtained with ClnDev.

Figures 3.3 and 3.4 provide some insights into the ASR performance under reverberation. Firstly, the correct classification rate per phoneme (main diagonal of the confusion matrices) clearly shows that the correct recognition rate significantly drops when reverberation is present, especially with pauses (/sil/) due to the time smearing of previous phonemes into these low energy gaps. Secondly, the distribution of the insertions (i.e. last row in both figures) is almost equally distributed for all phonemes and is similar under

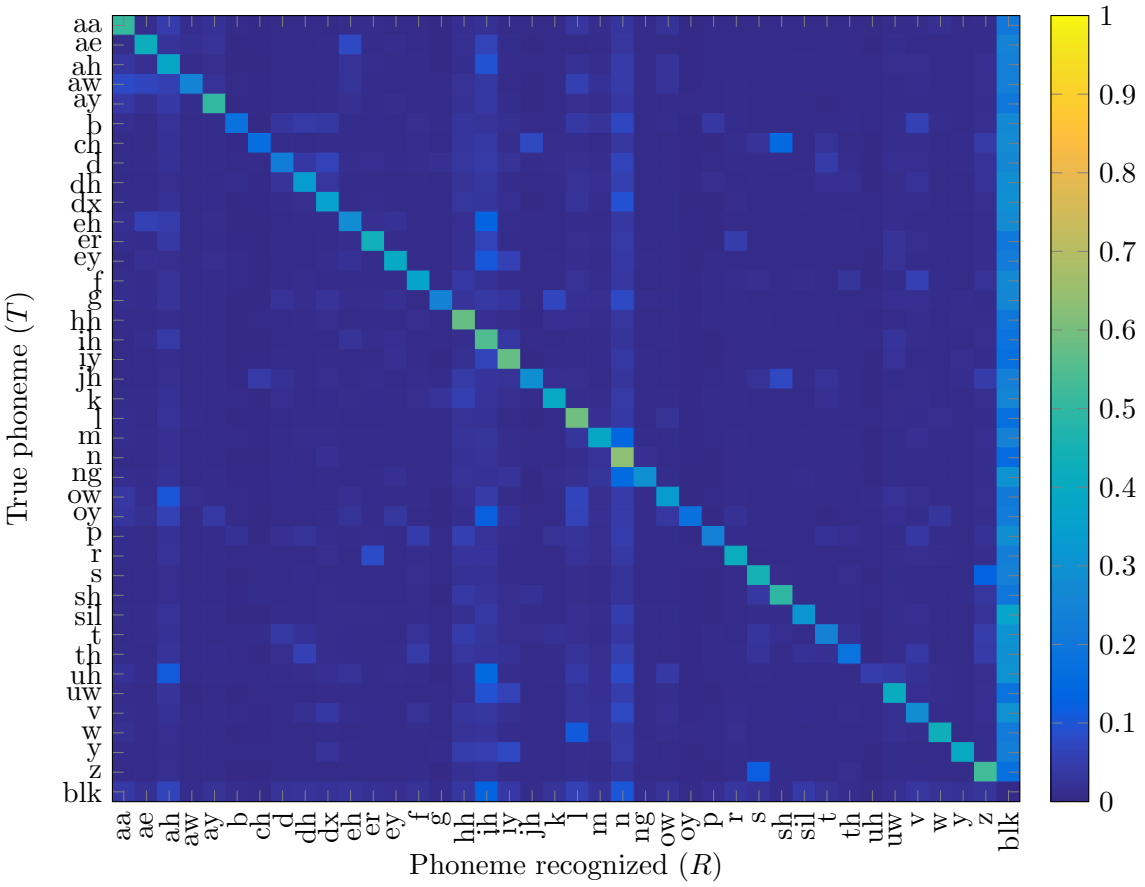

Figure 3.4: Phoneme confusion matrix obtained with RevDev.

non-reverberant and reverberant conditions. Thirdly, a considerable increase of deletions appears in RevDev as compared to ClnDev owing to time smearing which makes some phonemes to be recognized as the previous one. Finally, some phonemes are frequently confused, as for example phoneme /hh/, which is shown in the confusion matrix with vertical patterns of high values. This observation is in accordance with the conclusion presented in [126].

Table 3.2 displays the relative difference of the argument (rΔ) of correctly recognized (N_{cor}/N_{phn}), inserted (N_{ins}/N_{phn}), deleted (N_{del}/N_{phn}) and substituted (N_{sub}/N_{phn}) phoneme rate between ClnDev and RevDev computed as,

$$\mathrm{r}\Delta X = \frac{X_{\mathrm{ClnDev}} - X_{\mathrm{RevDev}}}{X_{\mathrm{ClnDev}}}, \tag{3.8}$$

where X can be N_{cor}/N_{phn}, N_{ins}/N_{phn}, N_{del}/N_{phn} or N_{sub}/N_{phn}.

As expected, the rate of correctly recognized phonemes decreases whereas deletions and substitutions are considerably increased under reverberation. However the insertion rate is slightly reduced. Table 3.2 indicates that ASR performance degradation is mainly caused by deletions and substitutions.

	rΔ N_{cor}/N_{phn}	**rΔ** N_{ins}/N_{phn}	**rΔ** N_{del}/N_{phn}	**rΔ** N_{sub}/N_{phn}
CI HTK	0.41	0.24	-2.37	-0.57
CI Kaldi	0.47	0.66	-2.88	-0.26
CD Kaldi	0.44	0.56	-4.38	-0.54

Table 3.2: Relative difference of phonemes recognition rates between ClnDev and RevDev.

It is clear that reverberation affects phoneme recognition differently depending on the reverberation level and the phoneme. The following section aims to model the phoneme errors at the output of the ASR using the confusion matrix which depends on the reverberation level. Such a model would be useful for predicting possible errors or for assigning confidence values to the phonemes derived from the confusability factor. In practice, C_{50} can be blindly estimated by applying the methods presented in Section 2.5.

3.2.3 Confusability factor in a Bayesian framework

Let T_k denote the true phoneme and R_k the recognized phoneme where k represents the phoneme label index. In this section a set of N_{phn}= 39 phonemes is considered. The $\mathcal{CF}(T_k,R_k,\mathrm{C}_{50})$ based on the probability of correctly recognized phoneme index k for a given reverberation level (C_{50}) is proposed as follows

$$\begin{aligned}\mathcal{CF}(T_k, R_k, \mathrm{C}_{50}) &= 1 - p(T_k|R_k, \mathrm{C}_{50}) = \\ &= 1 - \frac{p(R_k|T_k,\mathrm{C}_{50})\cdot p(T_k)}{\sum_{l=1}^{N_{phn}+1} p(R_k|T_l,\mathrm{C}_{50})\cdot p(T_l)},\end{aligned} \tag{3.9}$$

where the prior probability of the phoneme label T_k is $p(T_k) = \frac{\sum_{i=1}^{N_{phn}+1} N_{T_k R_i}}{\sum_{i=1}^{N_{phn}+1}\sum_{j=1}^{N_{phn}+1} N_{T_i R_j}}$, the likelihood of classifying the phoneme R_k given the phoneme label T_k and the reverberation

level C_{50} is $p(R_k|T_k, C_{50}) = \frac{N_{T_k R_k}}{\sum_{i=1}^{N_{phn}+1} N_{T_k R_i}}$, and $N_{T_k R_k}$ represents the number of times the phoneme label T_k is classified as R_k for a given C_{50}. It can be shown that the confusability factor presented in (3.9) can be computed directly from the confusion matrix as follows,

$$\begin{aligned}
\mathcal{CF}(T_k, R_k, C_{50}) &= 1 - \frac{p(R_k|T_k,C_{50})\cdot p(T_k)}{\sum_{l=1}^{N_{phn}+1} p(R_k|T_l,C_{50})\cdot p(T_l)} \\
&= 1 - \frac{\frac{N_{T_k R_k}}{\sum_{i=1}^{N_{phn}+1} N_{T_k R_i}} \cdot \frac{\sum_{i=1}^{N_{phn}+1} N_{T_k R_i}}{\sum_{i=1}^{N_{phn}+1}\sum_{j=1}^{N_{phn}+1} N_{T_i R_j}}}{\sum_{l=1}^{N_{phn}+1} \frac{N_{T_l R_k}}{\sum_{i=1}^{N_{phn}+1} N_{T_l R_i}} \cdot \frac{\sum_{i=1}^{N_{phn}+1} N_{T_l R_i}}{\sum_{i=1}^{N_{phn}+1}\sum_{j=1}^{N_{phn}+1} N_{T_i R_j}}} \\
&= 1 - \frac{N_{T_k R_k}}{\sum_{l=1}^{N_{phn}+1} N_{T_l R_k}}.
\end{aligned} \tag{3.10}$$

The phoneme indexes cover the range from 1 to $N_{phn}+1$ for the purpose of including, in addition to the substitution errors, the insertions and deletions in the computation of the confusability factor.

3.2.4 Results

Figure 3.5 illustrates the confusability factor presented in (3.10) with CD-KALDI for each recognized phoneme R_k (rows) at different levels of reverberation as measured using C_{50} (columns). It shows that the phoneme confusion is different for each phoneme and strongly depends on the reverberation level. In all cases, the confusability factor tends to increase when reverberation level increases. However, the rate of change varies significantly between phonemes. Similar behaviour of the confusability factor can be observed in Fig. 3.6 and Fig. 3.7 for CI-HTK and CI-KALDI respectively.

Figures 3.8, 3.9, 3.10 show the confusability factor achieved for each phoneme recognizer when the 39 phonemes considered previously are classified into 6 broad phoneme classes. The phonemes are clustered into 6 different classes of phonemes based on speech production manner [127]. The confusability factors obtained show that weak fricative phonemes (/th,v,hh,f,dh/) belong to the most confused class. On the contrary, silence broad phone class, which includes only /sil/ (pause), preserves a low confusability value amongst different reverberation levels due to the lack of energy of this broad phone class. Furthermore, the confusability factor magnitudes suggest that these phoneme confusions

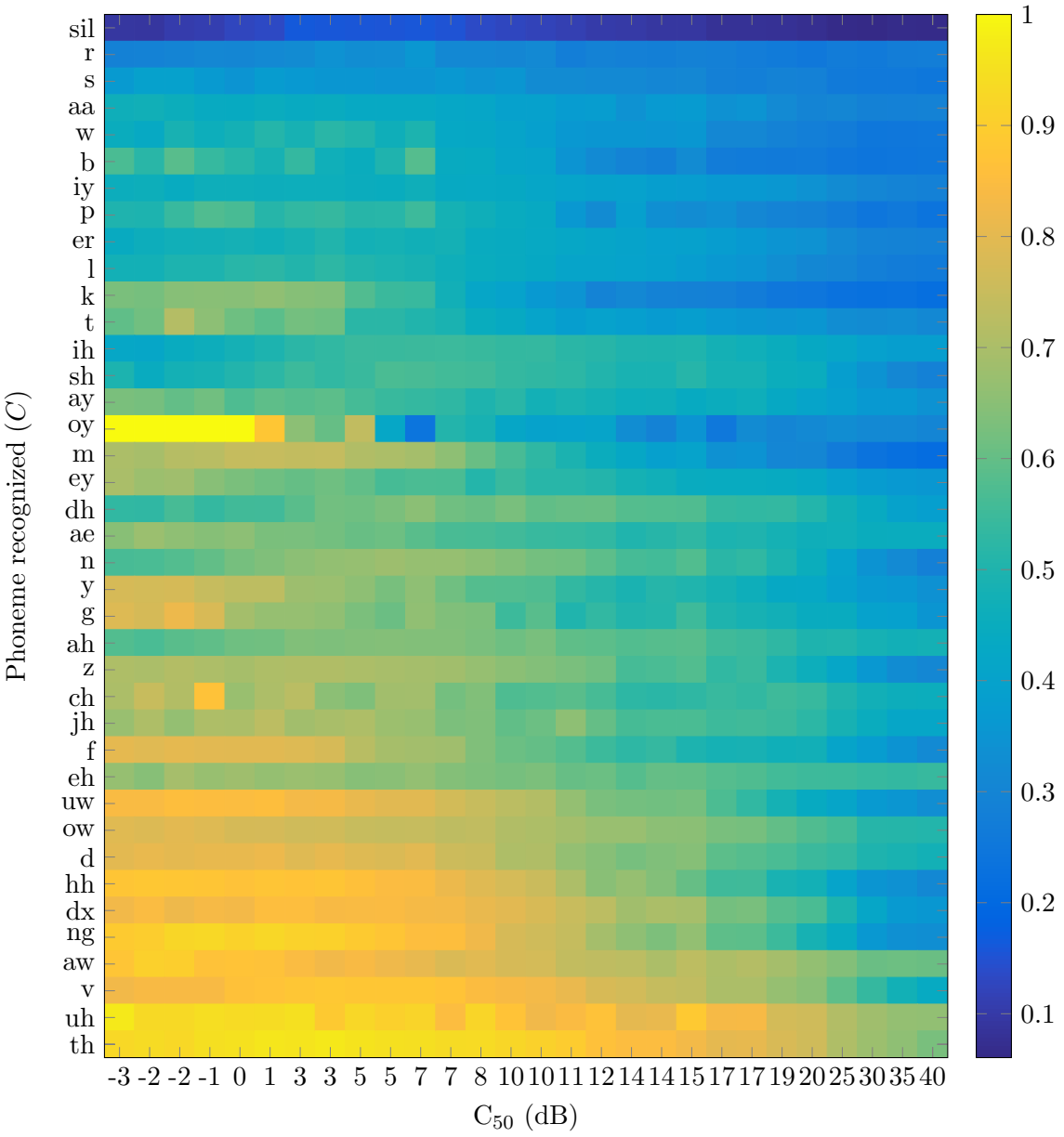

Figure 3.5: Confusability factor of the 39 phonemes for CD-KALDI with RevDev.

are similar for different phoneme recognizers.

In order to assess the repeatability of these results, the $\mathcal{CF}(T_k,R_k,\mathrm{C}_{50})$ computed from unseen RevEval data is compared to $\widehat{\mathcal{CF}}(T_k,R_k,\mathrm{C}_{50})$ a polynomial function fitted to RevDev. One polynomial function is computed for each phoneme or broad phone class and this function depends only on the C_{50} value. Thus, the confusability factor can be

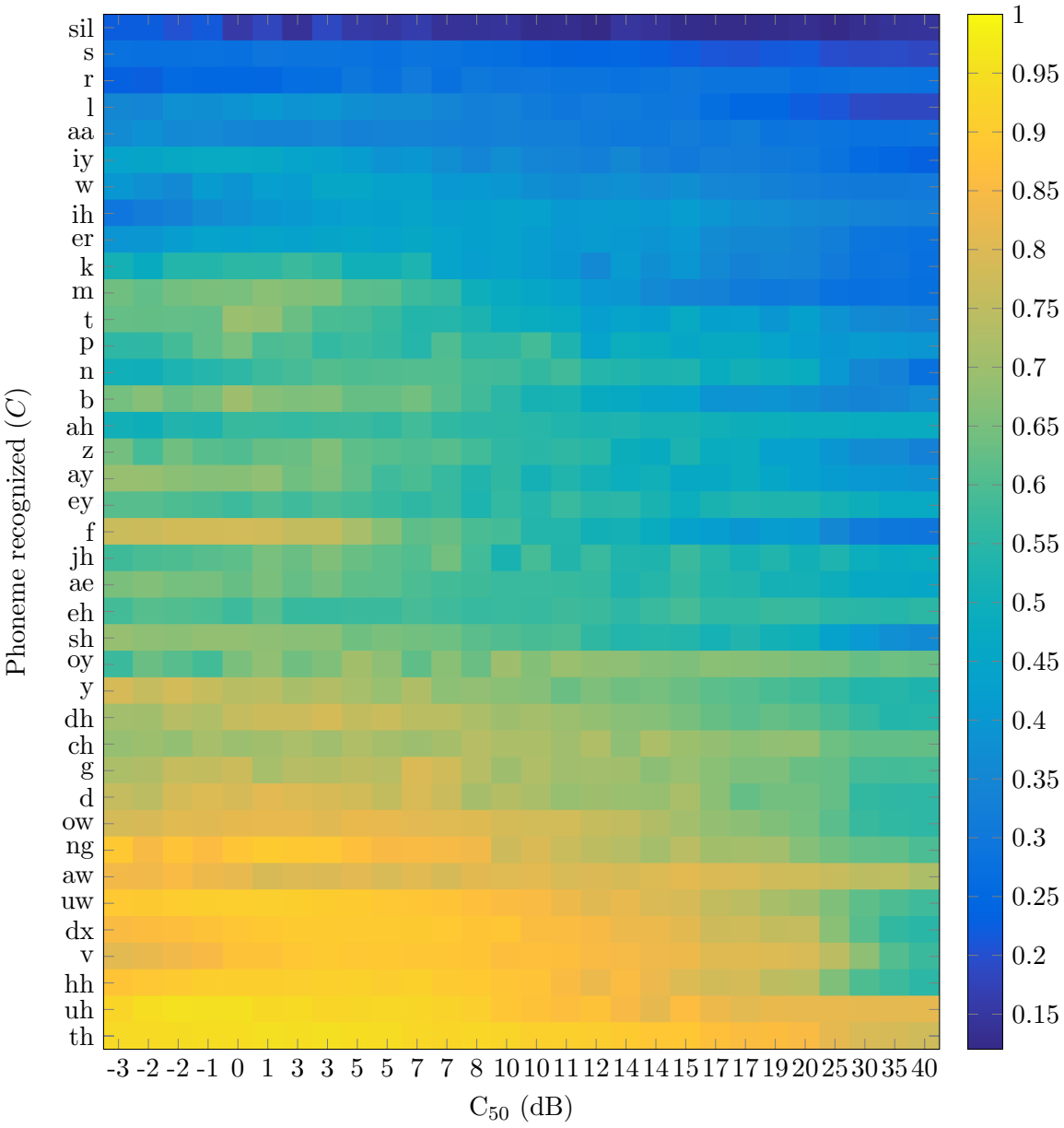

Figure 3.6: Confusability factor of the 39 phonemes for CI-HTK with RevDev.

extrapolated for any value of C_{50} using this polynomial function. A third order polynomial function fitted to $\mathcal{CF}(T_k,R_k,C_{50})$ for each phoneme k is used. The degree of polynomial was chosen such that the function minimizes the average Root Mean Square Deviation

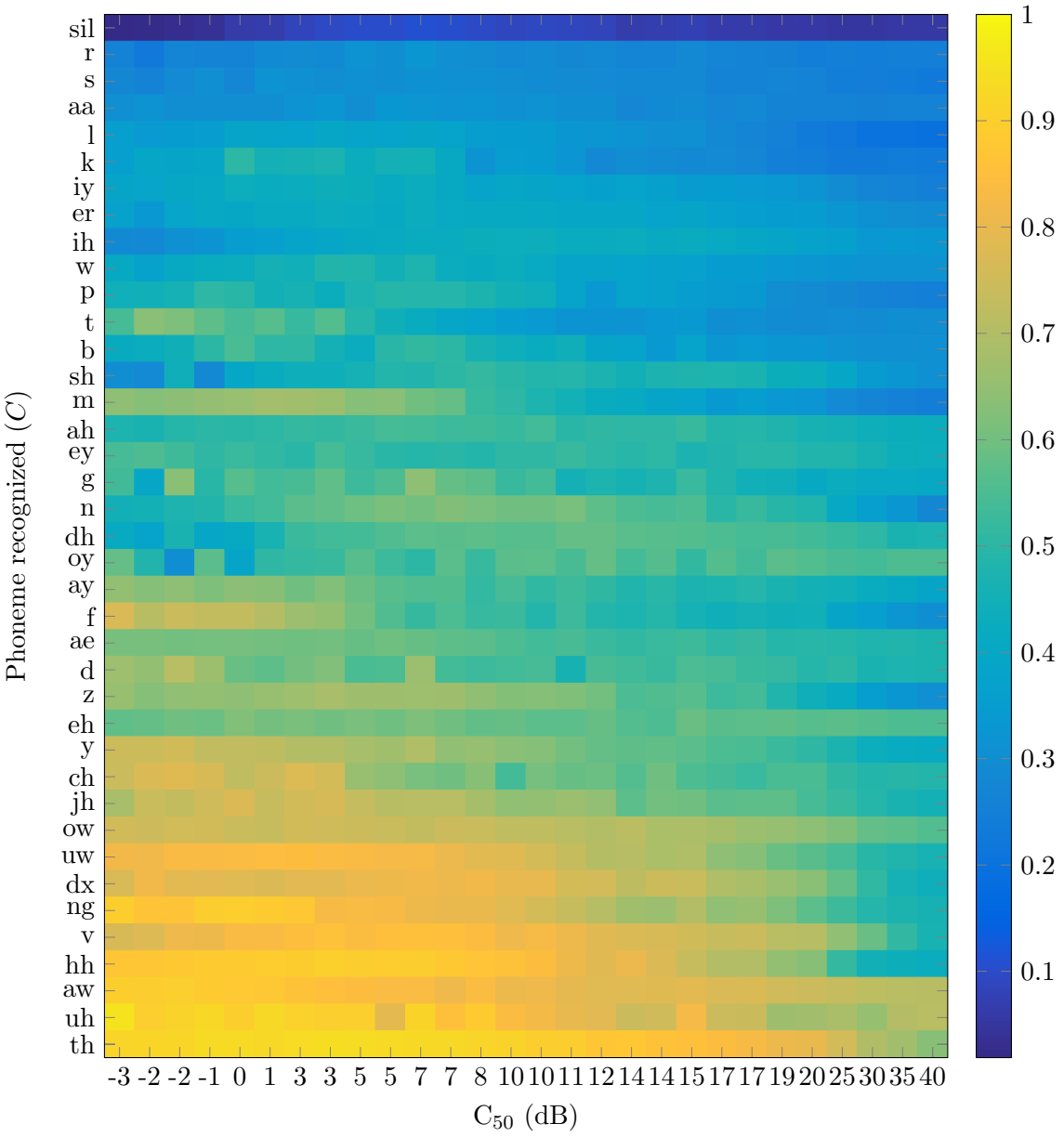

Figure 3.7: Confusability factor of the 39 phonemes for CI-KALDI with RevDev.

(aRMSD) in RevEval. The aRMSD is computed as follows,

$$\text{aRMSD} = \frac{1}{N_{phn} \cdot N_{cnd}^{1/2}} \sum_{k=1}^{N_{phn}} \sqrt{\sum_{c=1}^{N_{cnd}} \left(\widehat{\mathcal{CF}}(T_k, R_k, \text{C}_{50,\text{c}}) - \mathcal{CF}(T_k, R_k, \text{C}_{50,\text{c}}) \right)^2} \text{ dB}, \quad (3.11)$$

where N_{cnd} is the number of different reverberant conditions (i.e. different C_{50} values considered in the reverberant sets), and $\widehat{\mathcal{CF}}(T_k,R_k,\text{C}_{50})$ and $\mathcal{CF}(T_k,R_k,\text{C}_{50})$ are the fitted

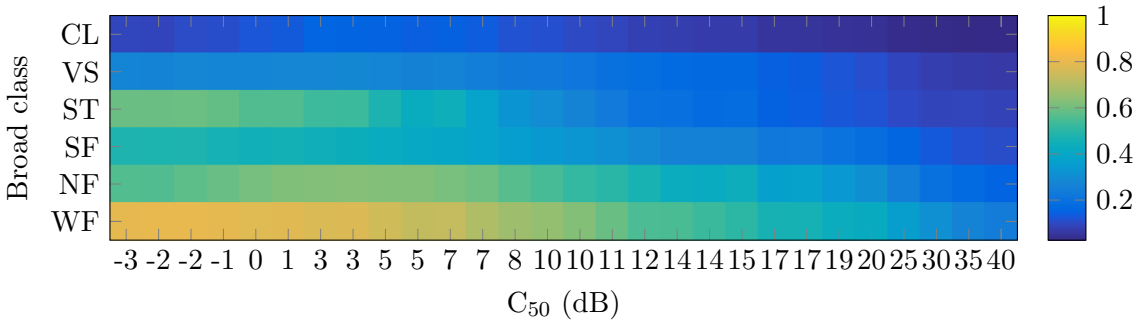

Figure 3.8: Confusability factor of 6 broad phone classes (Vowel/Semivowel (VS); Nasal/Flap (NF); Strong Fricative (SF); Weak Fricative (WF); Stop (ST); Closure (CL)) for CD-KALDI with RevDev.

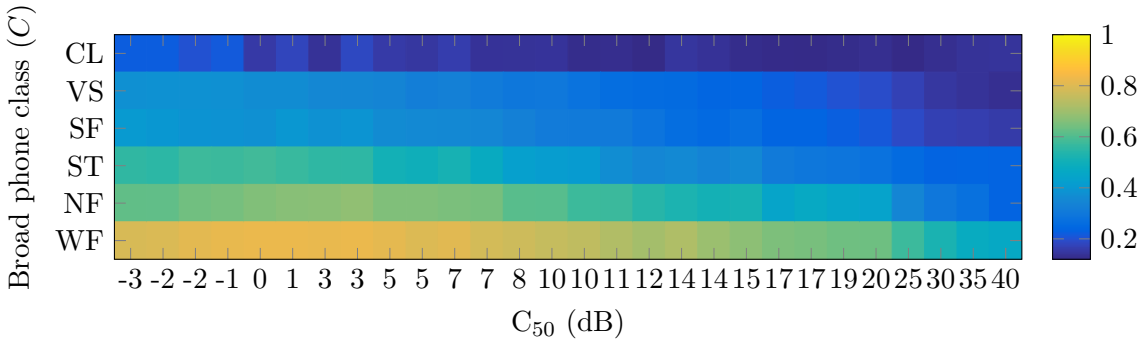

Figure 3.9: Confusability factor of 6 broad phone classes (Vowel/Semivowel (VS); Nasal/Flap (NF); Strong Fricative (SF); Weak Fricative (WF); Stop (ST); Closure (CL)) for CI-HTK with RevDev.

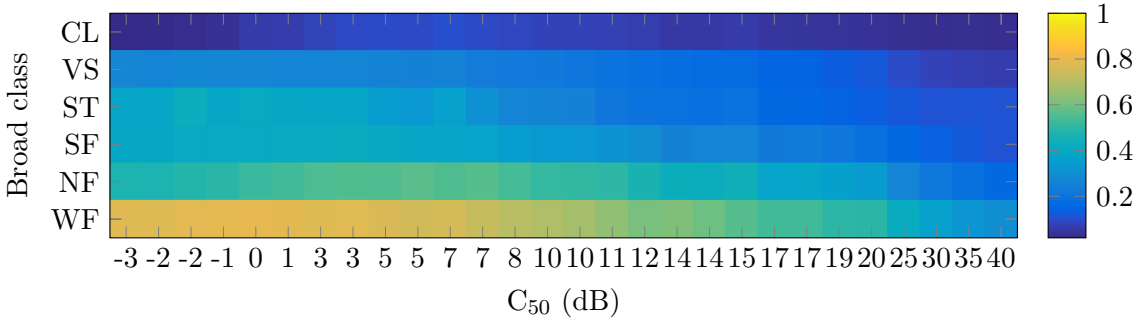

Figure 3.10: Confusability factor of 6 broad phone classes (Vowel/Semivowel (VS); Nasal/Flap (NF); Strong Fricative (SF); Weak Fricative (WF); Stop (ST); Closure (CL)) for CI-KALDI with RevDev.

function output and the confusability factor respectively for a given phoneme index k and reverberant condition index c.

Table 3.3 presents the aRMSD for RevDev and RevEval using a third order polynomial fitted to RevDev. It shows consistently low deviations for the three ASR configurations. As expected, the error in RevDev is lower because the polynomial function is fitted to this data but the error in RevEval still remains correspondingly low.

	CI-HTK	CI-KALDI	CD-KALDI
RevDev	0.030	0.037	0.035
RevEval	0.060	0.075	0.079

Table 3.3: The aRMSD achieved with a third order polynomial fitted on the confusability factors of 39 phonemes.

Table 3.4 shows the aRMSD obtained with the 6 broad phone classes displayed in Figs. 3.8, 3.9 and 3.10. The deviation is slightly decreased compared to using the 39 phonemes, however the relative difference between RevDev and RevEval remains approximately the same.

	CI-HTK	CI-KALDI	CD-KALDI
RevDev	0.025	0.023	0.025
RevEval	0.053	0.042	0.046

Table 3.4: The aRMSD achieved with a third order polynomial fitted on the confusability factors of the 6 broad phone classes.

Since RevEval comprises a completely independent set of RIRs (including real impulses responses) and recordings from RevDev, it is possible to conclude that a set of functions can be used to estimate a confusability factor of the recognized class under completely new reverberant environments. This model depends on C_{50}, apart from the ASR output R_k, which can be estimated employing external methods presented in Section 2.4.

3.2.5 Conclusions

The degradation in phoneme recognition was analyzed under reverberation with different speech recognition toolkits, i.e. Hidden Markov Model Toolkit (HTK) and Kaldi. This analysis showed that, for ASR, phonemes vary in their robustness to reverberation. The

confusion matrix presented indicates the ASR robustness of each phoneme to different levels of reverberation. It has also been shown that the main errors in our tests are deletions and substitutions. Motivated by these observations, a metric that characterizes the confusion of recognizing the phoneme in a Bayesian framework is proposed. Finally, the results of the experiments have demonstrated that for a strongly reverberant scenario with $C_{50} = 6$ dB, the most robust phoneme is /r/ whereas the most fragile phonemes are the class of weak fricatives (e.g. TIMIT phonetic label /hh,th,v/).

3.3 Reverberant speech recognition using the confusability factor

The confusability factor investigated in Section 3.2 suggests that some acoustic units are more confusable than other acoustic units, and this difference increases with the level of reverberation present in the signal. In this section a method to increase robustness of the ASR against reverberation by scaling the acoustic probabilities according to the $1 - \mathcal{CF}(T_k,R_k,C_{50})$ is described. This technique can be classified as an uncertainty-based approach [128], where the uncertainty is modelled with the confusability factor. The potential of this approach is illustrated on the data made available for the ASR task of the previous REVERB Challenge [74] which was launched by the IEEE Audio and Acoustic Signal Processing Technical Committee in order to compare ASR performance on a common data set of reverberant speech.

3.3.1 Method

The aim of this method is to perform reverberant speech recognition by employing the confusability factor to scale the likelihoods $p(\mathbf{O}|\boldsymbol{\mathcal{W}})$ in (3.5). Thus, the ASR maximization formulation is

$$\arg\max_{\boldsymbol{\mathcal{W}}} p(\mathbf{O}|\boldsymbol{\mathcal{W}})p(\boldsymbol{\mathcal{W}}), \tag{3.12}$$

where the acoustic probability $p(\mathbf{O}|\boldsymbol{\mathcal{W}})$ is scaled following

$$p(\mathbf{O}|\boldsymbol{\mathcal{W}}) = \sum_{\mathcal{S}} \pi(\mathcal{S}_0) \prod_{l=1}^{N_{frm}} a_{\mathcal{S}_{l-1}\mathcal{S}_l} \prod_{l=0}^{N_{frm}} p(\mathcal{S}_l|\mathbf{o}_l)(1 - \mathcal{CF}(T_{\mathcal{S}_l}, R_{\mathcal{S}_l}, \mathrm{C}_{50})). \tag{3.13}$$

This method is implemented internally in the HTK decoder software such that it modifies the acoustic probability $p(\mathbf{O}|\mathbf{W})$ as it is computed. Alternatively, expression 3.12 can be approximated by re-scoring the ASR output hypotheses such as N-best lists of lattices, however this is not investigated in the thesis.

3.3.1.1 Illustrative example of the proposed method

For the purpose of illustrating this method, a lattice obtained from a TIMIT recording employing CD-KALDI is shown in this section.

A lattice is a structure that comprises nodes and arcs, where the nodes can contain time information while arcs are associated with symbols and scores. Lattices represent ASR multiple hypotheses in a compact manner. In Kaldi, the lattices are represented with Weighted Finite-State Transducers (WFST) [129] where weights correspond to the likelihoods, input symbols are the transition-id from the HMMs and output symbols are typically words or phonemes depending on the recognition type. The nodes in this case contain only an unique identifier. Figure 3.11 shows an example of a lattice obtained by phonetically recognizing reverberant speech. This lattice is a small segment of the complete lattice obtained after recognizing the reverberant utterance "Medieval society was based on hierarchies". The correct path is shown in blue while red path represents the recognized path. In this particular example there are three arcs that are incorrectly recognized. Even though the output symbols of the correct path are in some cases the same as the incorrect path (e.g. /s, dx, iy/), lattices have only one unique path from the beginning to the end, therefore in this case the full path is incorrect. The aim is to scale the likelihoods of the arcs such that the correct path is finally recognized.

For example, at node 12 in Fig. 3.11, the likelihood of the arch from node 12 to node 13 should be severely reduced since /dx/ is a phoneme with high confusability factor

(Fig. 3.5) whereas the likelihood of the arch from node 12 to node 378 should not be severely reduced since /ih/ has a low confusability factor. Therefrom, the most likely path in the lattice is amended to follow the correct path.

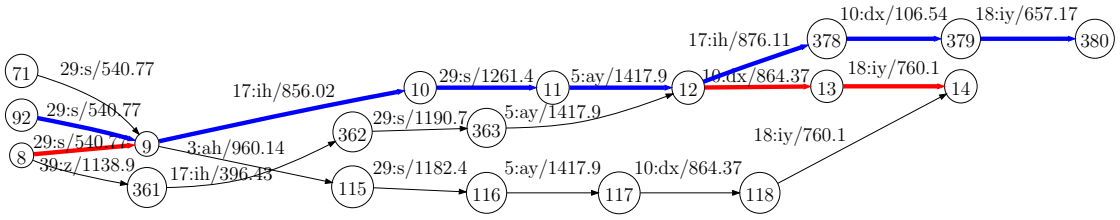

Figure 3.11: Extracted segment of the lattice obtained when employing ASR on the reverberant ($C_{50} \approx 20$ dB) TIMIT utterance "Medieval society was based on hierarchies". Arcs are labelled with the format transition-id:phoneme/likelihood. This segment of the lattice belongs to the word "society". Red path corresponds to the most probable path and the correct recognition path is represented in blue.

3.3.2 Experimental setup

This method is evaluated within the REVERB Challenge framework. The REVERB Challenge database comprises three different sets: training set that includes simulated reverberant speech; development set that includes simulated and real recordings; and evaluation set that contains simulated and real recordings as well. The training set is employed to train the ASR models. The development set is used to learn the confusability factor following (3.10) whereas the method presented in (3.12) is evaluated on the evaluation set. Regarding the RIRs employed to create each set, on the one hand the simulated RIRs are obtained in three different room sizes (small, medium and large) at two different microphone-source positions (near and far). Section 3.4.2 analyses these room impulse responses in terms of C_{50}. On the other hand, real reverberant recordings are captured in one room at two different microphone-source positions. Additionally, stationary background noise is added to simulated reverberant utterances and the real reverberant utterances contain also noise present during the recording.

In these experiments a perfect estimation of the room characteristics and position in the room is assumed, therefore the phoneme confusability factor for each position and room type in the development set is computed first and then this confusability factor is

applied to the equivalent positions and room types in the evaluation set.

A total of 44 confusability factors for each evaluation subset is computed following the phone set of the BEEP dictionary[3]. Additionally, two more acoustic units are added, silence and short pause, which are set to zero, in accordance with Fig. 3.10, due to the lack of these tokens in the transcriptions.

The evaluation metric used to assess the performance of this approach is PER (2.6), which can be rewritten as,

$$\text{PER} = \frac{N_{del} + N_{ins} + N_{sub}}{N_{del} + N_{cor} + N_{sub}} \tag{3.14}$$

where N_{cor} is the number of phonemes correctly recognized, N_{del} is the number of deletions, N_{sub} is the number of substitutions and N_{ins} is the number of insertions. In addition to PER, also these last 4 elements are analysed.

The baseline employed for comparison purposes is the REVERB Challenge HTK ASR trained on clean data and modified to perform context-dependent phoneme recognition. It uses MFCC features including Delta and Delta-Delta coefficients and tied-state HMM acoustic models with 10 Gaussian components per state. The proposed method employs the exact same configuration as this baseline, the only difference is the scaling of the acoustic probabilities.

3.3.3 Results

Figure 3.12 shows the PER for each of the evaluation subsets. The proposed method reduces the PER for every subset included in the REVERB Challenge. The maximum error reduction is accomplished on the two sets with real recordings where the PER is reduced by more than 4%. The overall absolute PER reduction achieved with this method is 3.2%.

In Table 3.5 the error is broken down into the four different elements that are involved in the PER computation (3.14). The rate of phonemes correctly recognized is on

[3] ftp://svr-ftp.eng.cam.ac.uk/pub/comp.speech/dictionaries/beep.tar.gz

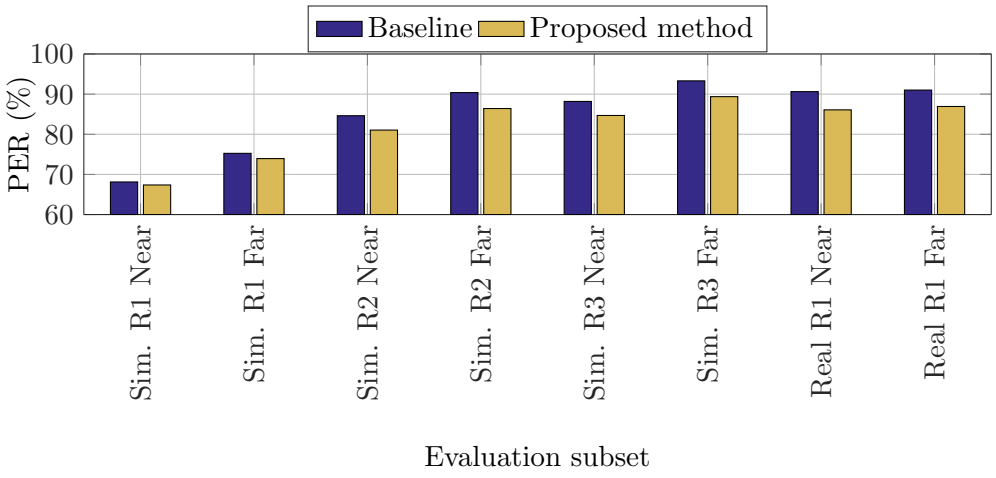

Figure 3.12: Comparison between the PER (%) obtained with the baseline system and the PER (%) achieved with the proposed method using the confusability factor.

average 0.8% higher with the proposed method in this section, however it is not consistently higher than the baseline. The proposed method provides lower substitution and insertion phoneme rate than the baseline for all subsets. On average, the reduction achieved is 5.7% and 2.4% respectively. On the contrary, the deletion phoneme rate achieved with the proposed method is higher than the baseline, on average 4.9%.

Recording type			N_{cor}/N_{phn}		N_{sub}/N_{phn}		N_{ins}/N_{phn}		N_{del}/N_{phn}	
			Bas.	**Prop.**	**Bas.**	**Prop.**	**Bas.**	**Prop.**	**Bas.**	**Prop.**
Sim.	*R1*	*near*	63.68	62.28	27.97	27.10	31.81	29.65	8.34	10.62
		far	58.16	57.27	34.26	32.01	33.41	31.21	7.58	10.72
	R2	*near*	43.80	42.91	37.71	32.72	28.42	23.95	18.49	24.37
		far	32.04	33.43	48.25	39.77	22.42	19.84	19.71	26.80
	R3	*near*	44.36	42.63	43.25	37.12	32.54	27.32	12.38	20.25
		far	31.62	33.68	52.94	44.14	24.92	23.05	15.44	22.18
Real	*R1*	*near*	27.54	31.49	52.50	45.54	18.16	17.57	19.96	22.97
		far	30.21	34.07	56.10	49.26	21.21	20.99	13.69	16.67

Table 3.5: Comparison between the correctly recognized (N_{cor}/N_{phn}), substituted (N_{sub}/N_{phn}), inserted (N_{ins}/N_{phn}) and deleted (N_{del}/N_{phn}) phoneme rate, achieved with the baseline (Bas.) and with the modified recognition using the confusability factor (Prop.).

The increment of deletions and the reduction of substitutions and insertions indicates that the proposed method removes incorrect phonemes from the ASR hypothesis, and furthermore it slightly improves the number of correctly recognized phonemes. Con-

sequently the overall PER is reduced.

In addition to the phoneme recognition, a word recognition is performed. The improvement achieved in this case is significantly lower; only 0.18% absolute Word Error Rate (WER) reduction with respect to the baseline WER 48.04%. This may be due to the incorporation of the language model and dictionary rules into the ASR which is solving some of the problems present in reverberant environments.

3.3.4 Conclusions

A method to improve the phoneme recognition in reverberant environment has been presented. This method employs the confusability factor to scale down the acoustic probabilities of less robust phonemes during recognition according to the confusability factor.

The method has been evaluated on the REVERB Challenge showing that in general it removes incorrect phonemes from the ASR hypothesis while preserving correct phonemes which reduces the final phoneme error rate. The total reduction in PER achieved with this approach on REVERB Challenge compared to the REVERB Challenge baseline is 3.2%. Despite this improvement, the PER remains still high as shown in Fig. 3.12 which indicates that this approach needs to be combine with other approaches to be useful from the application point of view.

3.4 Reverberant speech recognition using C_{50}

The method proposed in this section is a hybrid approach based on front-end-based and back-end-based single-channel techniques. The C_{50} estimate is employed to select different acoustic models (back-end approach) which are trained on feature vectors appended to include the C_{50} value (front-end approach). The resulting appended feature vector is then reduced in dimension to match the original dimensionality by applying Heteroscedastic Linear Discriminant Analysis (HLDA) [130]. The technique was tested within the ASR task of the REVERB Challenge [74] as in previous section.

3.4.1 C_{50} estimator

Two different single-channel C_{50} estimators are employed in this work: Non-Intrusive Room Acoustic estimation using Classification And Regression Trees (NIRA-CART) presented in Section 2.3.2.1 with input feature as in [12] and Non-Intrusive Room Acoustic estimation using bidirectional long-short term memory (NIRA-BLSTM) described in Section 2.3.2.4. The motivation for using NIRA-CART described in [12] is to show the impact of the C_{50} accuracy in this reverberant speech recognition method. In this work C_{50} is used to characterize reverberation in the signal instead of T_{60} as in [107] because this last measure is independent of the source-receiver distance which is a key factor in the speech degradation. Moreover C_{50} was shown to be highly correlated with the ASR performance compared to other measures of reverberation [12] [27] which makes it suitable for this purpose.

3.4.1.1 Wide-band feature set extension

These C_{50} estimators were originally proposed to operate on speech signals sampled with a sampling frequency of 8 kHz and extended in Section 2.6.2 to 16 kHz speech signals. In this section the latter configuration is employed, therefore the feature vector per utterance for NIRA-CART comprises 393 elements while the feature vector per frame for NIRA-BLSTM includes 134 features.

3.4.2 Analysis of the challenge data

The database provided in REVERB Challenge comprises 3 different sets of 8-channel recordings: training set, development set and evaluation set. Real data recorded in a reverberant room and simulated data created by convolving non-reverberant utterances with measured RIRs are included in development set and evaluation set whereas training set only comprises simulated data. This section analyses the RIRs of different data sets in terms of C_{50} inasmuch as this is a key aspect in the design of the proposed algorithms.

Figure 3.13 shows the histogram of C_{50} values for the 24 training RIRs[4] including all channels of each response. As seen in Fig. 3.13, the RIR training set covers a wide range of C_{50} spanning approximately 25 dB. These RIRs are used to create the data set employed to train our C_{50} estimators by convolving these RIRs with speech signals from the training set which, for the REVERB Challenge, was formed from the Wall Street Journal recorded at the University of CAMbridge phase 0 (WSJCAM0) training set [131].

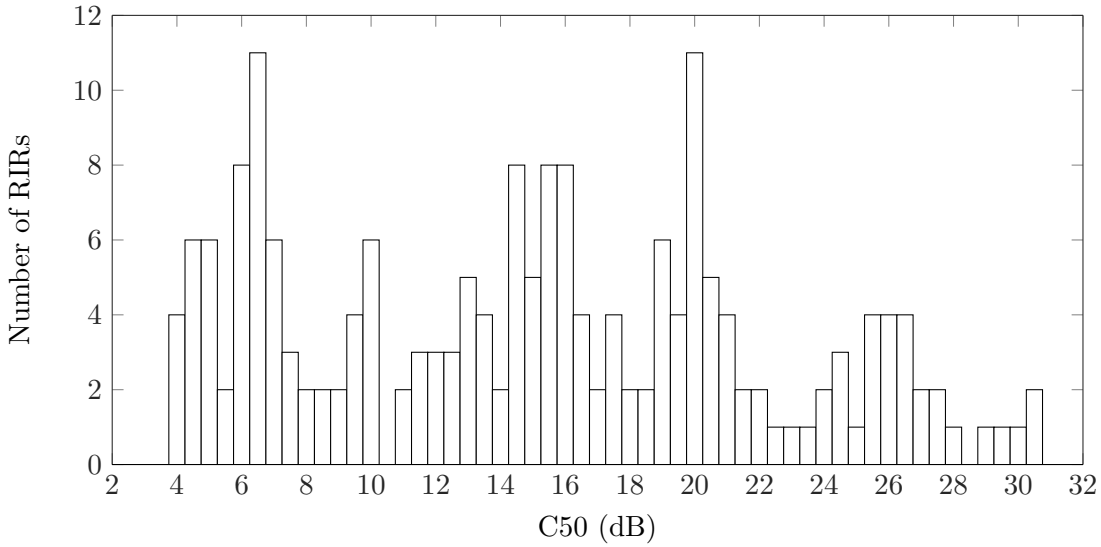

Figure 3.13: Histogram of C_{50} values in the training set.

Table 3.6 presents the measured C_{50} of the RIRs included in the development and evaluation sets of simulated data[5]. It shows a significant difference between the small room recordings (Room1) which are less reverberant ($T_{60} = 0.25$ s), and the medium and large room recordings (Room2 and Room3 respectively) which have higher reverberation times ($T_{60} = 0.5$ s and $T_{60} = 0.7$ s respectively). Furthermore, the two distances of the speaker from the microphone, this is, *near* = 50 cm and *far* = 200 cm, show a constant C_{50} difference of 8 dB to 10 dB.

Real recordings are captured in a reverberant meeting room from two different distances: near (≈100 cm) and far (≈250 cm). The development and evaluation sets of these recordings are not analysed in terms of measured C_{50} since the RIRs of these sets

[4] http://reverb2014.dereverberation.com/tools/reverb_tools_for_Generate_mcTrainData.tgz
[5] http://reverb2014.dereverberation.com/tools/reverb_tools_for_Generate_SimData.tgz

		Room1 T60 = 0.25 s		**Room2 T60 = 0.5 s**		**Room3 T60 = 0.7 s**	
		near	*far*	*near*	*far*	*near*	*far*
Dev. set	C_{50} (dB)	30.78	21.62	16.52	7	16.37	6.69
Eval. set	C_{50} (dB)	29.44	22.04	14.47	6.27	15.10	7.06

Table 3.6: C_{50} measures of the RIRs included in the development set (Dev. set) and evaluation set (Eval. set) of the simulated data from the REVERB Challenge.

are unavailable.

3.4.2.1 C_{50} estimator performance

The evaluation metric used to compare the C_{50} estimator performance is the RMSD given as:

$$\text{RMSD} = \sqrt{\frac{1}{N_{utt}} \sum_{u=1}^{N_{utt}} (\widehat{\text{C}_{50,\text{u}}} - \text{C}_{50,\text{u}})^2} \text{ dB}, \tag{3.15}$$

where N_{utt} is the total number of utterances considered to compute the RMSD and $C_{50,u}$ and $\widehat{C_{50,u}}$ are the measured ground truth and estimated value respectively for the uth utterance.

The training set is randomly split into a training subset (80% of the data used to train the models) and evaluation subset (20% of the recordings employed to evaluate the models) in order to provide insights into the performance of both C_{50} estimators. Additionally, the performance of the C_{50} estimators is also evaluated using the development set and evaluation set of the simulated data whose C_{50} measures are presented in Table 3.6. Table 3.7 summarizes the RMSD performance of each estimator evaluated in these data sets. NIRA-BLSTM achieves the lowest deviation in each data set, providing on average a RMSD 1.57 dB lower than NIRA-CART. Both estimators exhibit lower deviations on the evaluation subset of the training set (i.e. *Training set - eval. subset*) because this reverberant subset is similar to the data used to train the C_{50} estimators.

Estimator	RMSD (dB)		
	Training set - eval. subset	*Sim. data - dev. set*	*Sim. data - eval. set*
NIRA-CART	1.86	3.60	3.16
NIRA-BLSTM	0.48	1.98	1.45

Table 3.7: RMSD of the C_{50} estimators tested in three different sets.

3.4.3 Methods

This section describes different configurations for reverberant speech recognition. The idea underpinning these methods is to exploit estimated C_{50} to improve robustness of ASR to reverberation. Section 3.4.3.1 introduces the front-end techniques, Section 3.4.3.2 describes the back-end methods and finally Section 3.4.3.3 presents the combination as outlined in Fig. 3.14.

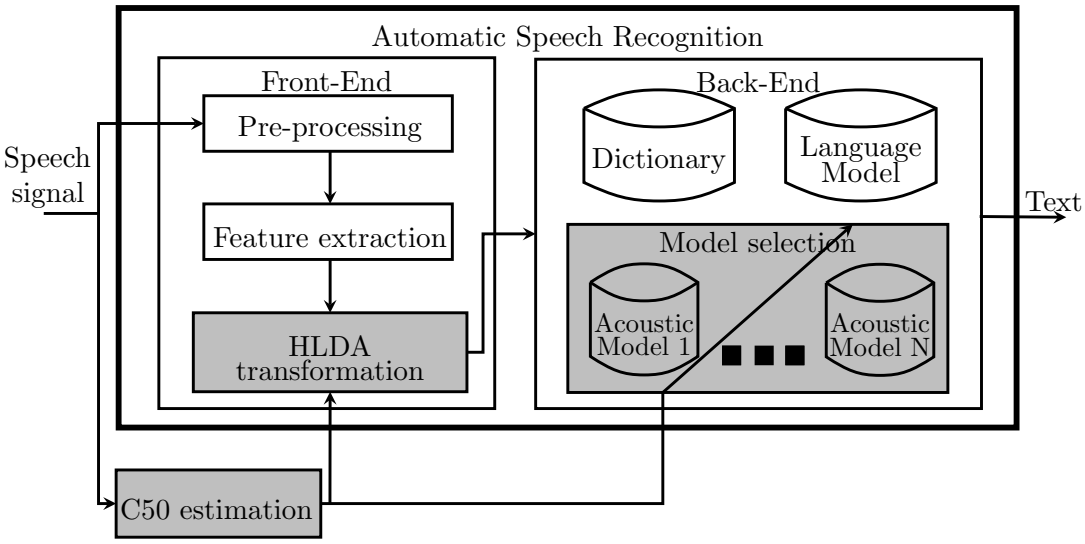

Figure 3.14: Reverberant speech recognition using C_{50} estimation.

3.4.3.1 C_{50} as a supplementary feature in ASR

In this approach, the estimated C_{50} is included as an additional feature in the ASR feature vector. The baseline recognition system uses a feature vector with 13 MFCCs, with the first and second derivatives of these coefficients followed by cepstral mean subtraction.

Two alternative improved configurations are now proposed. The first proposed

configuration (**C50FV**) is to add C_{50} estimation directly to this feature vector. Therefore the modified feature vector comprises 40 elements.

In a second configuration (**C50HLDA**) the feature vector dimension is reduced using Linear Discriminant Analysis (LDA) [59]. This method projects the input feature vector $\boldsymbol{o}_l$ of lth frame onto a new space $\bar{\boldsymbol{o}}_l$ by applying a linear transformation $\mathbf{W}$ such that

$$\bar{\boldsymbol{o}}_l = \mathbf{W}^T \boldsymbol{o}_l, \tag{3.16}$$

where $\mathbf{W}$ is an $q_r \times q_c$ matrix, q_r is the dimension of the input feature, i.e. 40, and q_c is the dimension of the transformed feature space, i.e. 39. This transformation in general retains the class-discrimination in the transformed feature space. The transformation $\mathbf{W}$ is obtained by maximizing the ratio of the between-class scatter matrix $\mathbf{S}_B$ to the within-class scatter matrix $\mathbf{S}_W$, this is,

$$\breve{\mathbf{W}} = \arg\max_{\mathbf{W}} \frac{\det\left(\mathbf{W}^T \mathbf{S}_B \mathbf{W}\right)}{\det\left(\mathbf{W}^T \mathbf{S}_W \mathbf{W}\right)}, \tag{3.17}$$

where det() represents the determinant of a matrix.

The projection that maximizes (3.17) corresponds to $\breve{\mathbf{W}}$ whose columns are the eigenvectors of ${\mathbf{S}_W}^{-1}\mathbf{S}_B$ with the q_c highest eigenvalues so that q_c is the dimension of the reduced feature space.

In this section a model-based generalization of LDA [130] is used. In this case the linear transformation is estimated from Gaussian models using the expectation-maximization algorithm. For these models it is assumed that class distributions with equal mean and variance across all classes do not contain discriminant classification information.

In all configurations, the acoustic models are trained using the modified feature space.

3.4.3.2 Model selection

The proposed back-end approach aims to select the optimal acoustic model $\breve{\mathcal{A}}$ such as:

$$\breve{\mathcal{A}}(\mathrm{C}_{50}) = \begin{cases} \mathcal{A}_1 & -\infty < \mathrm{C}_{50} \leq \psi_1 \\ \mathcal{A}_2 & \psi_1 < \mathrm{C}_{50} \leq \psi_2 \\ . & . \\ . & . \\ . & . \\ \mathcal{A}_{N_\mathcal{A}} & \psi_{N_\mathcal{A}-1} < \mathrm{C}_{50} < \infty \end{cases} \tag{3.18}$$

where $N_\mathcal{A}$ represents the number of available acoustic models $\boldsymbol{\mathcal{A}} = \{\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_{N_\mathcal{A}}\}$ and $\boldsymbol{\psi} = \{\psi_1, \psi_2, \cdots, \psi_{N_\mathcal{A}-1}\}$ is the vector with the C_{50} threshold values sorted in ascending order.

3.4.3.2.1 Model switching between REVERB Challenge acoustic models

The first configuration (**Clean&Multi cond.**) is based on selecting between the two baseline acoustic models, i.e. $N_\mathcal{A} = 2$, provided in the challenge (clean-condition HMMs and multi-condition HMMs) according to the level of C_{50} estimated from the input signal. In this case, $\mathcal{A}_1$ represents the multi-condition HMMs and $\mathcal{A}_2$ is the clean-condition HMMs. By empirical optimization over the development data set and considering the analysis carried out in Section 3.4.2, the model switching threshold $\psi_1 = 23$ dB is chosen. Therefore, input speech signals with estimated C_{50} higher than 23 dB are recognized using clean-condition HMMs whereas signals with C_{50} lower than this threshold are recognized using multi-condition HMMs.

3.4.3.2.2 Model switching using newly trained acoustic models

Second and subsequent configurations are now introduced based on training new reverberant acoustic models. The data set used to train the models is always the clean training set convolved with the training RIRs (Fig. 3.13). In order to include in the trained models $\boldsymbol{\mathcal{A}}$ all the representative data of the acoustic units (i.e. triphones), all N_{utt} clean training utterances are convolved with a subset of N_{RIR} training RIRs to create a reverberant

acoustic model $\mathcal{A}_a$ such as

$$y_u(n) = \sum_{m=0}^{M-1} h_u(m) s_u(n-m) \quad u = 1, 2, \cdots, N_{utt} \tag{3.19}$$

where y_u is the reverberant speech obtained with the clean utterance s_u and the RIR h_u in the row (u mod N_{RIR}) of the matrix $\boldsymbol{H}_a$. This matrix contains the N_{RIR} RIRs with a C_{50} value that satisfies $\psi_{a-1} < C_{50} \leq \psi_a$.

The first approach is to create three reverberant acoustic models (**MS3**) according to the C_{50} values of the RIRs as shown in Fig. 3.15(a). The threshold vector is set to $\boldsymbol{\psi} = \{10, 20\}$ dB, which was derived from the C_{50} estimations of the development set shown in Table 3.6 such that each model covers 2 different reverberant conditions. The aim is to cluster the development set into three groups with similar ASR performance and train a model for each group. The most reverberant model $\mathcal{A}_1$ is trained with the RIRs that have C_{50} lower than 10 dB, then the second acoustic model $\mathcal{A}_2$ is trained with RIRs that have C_{50} between 10 dB and 20 dB, and finally the third model $\mathcal{A}_3$, which represents the least reverberant conditions, is trained with those RIRs with a C_{50} higher than 20 dB.

Next configuration (**MS5**) includes the use of classes with overlapping ranges of C_{50} in order to build the acoustic models. For each class, the overlap range of C_{50} was approximately 50% of the size of the neighbouring class. This configuration results in the same previous models (MS3) but adds two additional models spanning the transitional ranges of C_{50}. These two models provide a smoother transition between acoustic models. The acoustic model most representative of reverberation level estimated from the utterance is selected in the recognition phase. Figure 3.15(b) shows the construction of MS5 during training and the thresholds used to select models in the recognition stage.

Additional configurations were tested by increasing the number of models trained: 8 overlapped acoustic models (**MS8**), 11 overlapped acoustic models (**MS11**), 14 overlapped acoustic models (**MS14**) and 18 overlapped acoustic models (**MS18**). These models are obtained by further dividing the original MS3 configuration. By increasing the number of models the range of C_{50} of the training data of each model is decreased in terms of C_{50} which creates acoustic models more specific for each reverberant condition. Figure 3.16

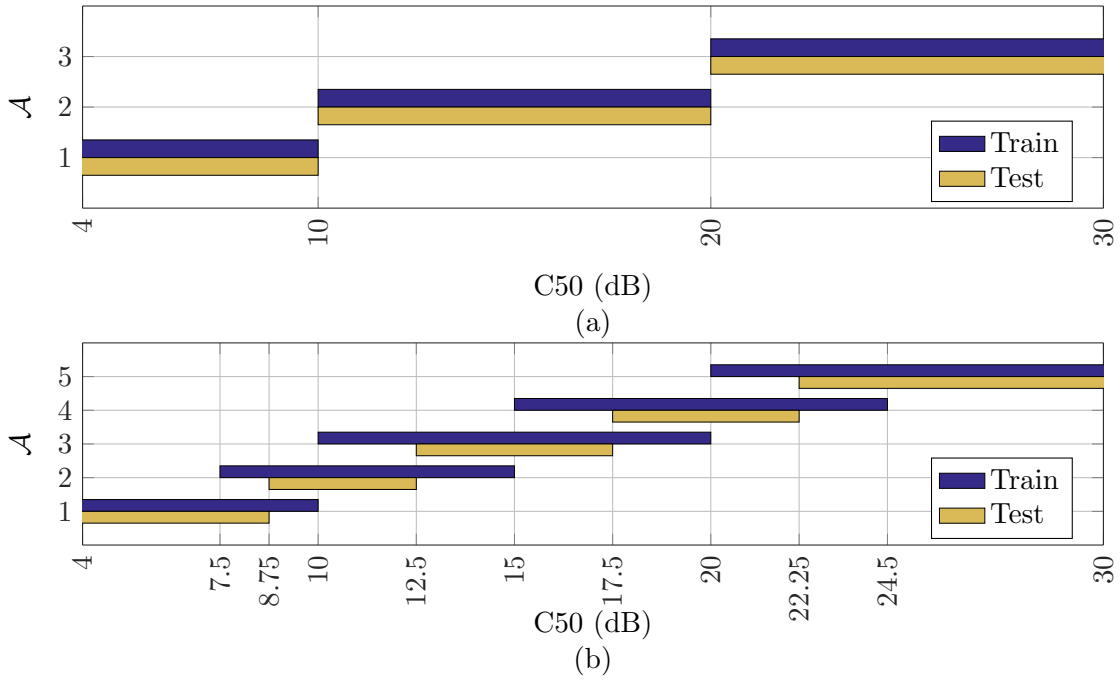

Figure 3.15: Comparison of MS3 (a) and MS5 (b) configurations for training the acoustic (blue bars) models and recognizing testing data (light brown bars) according to C_{50}. The difference relies on the overlapping of the training data for MS5 configuration.

shows the ranges of C_{50} used for MS11.

3.4.3.3 Model selection including C_{50} in the feature vector

This method combines the two approaches described above: C50HLDA and model selection. Figure 3.14 shows the block diagram of this method where grey modules represent the modifications included to design this method. Firstly, C_{50} is estimated from the speech signal. This C_{50} estimate is then included in the feature vector before applying the HLDA transformation and also used to select the most suitable acoustic model.

All the tested configurations employ the C_{50} thresholds as described in Section 3.4.3.2 to create the data to train the acoustic models and to select the appropriate acoustic model in the recognition stage. These configurations are referred as **MS**$N_{\mathcal{A}}$**+ C50HLDA**, where $N_{\mathcal{A}}$ represents the number of acoustic models created.

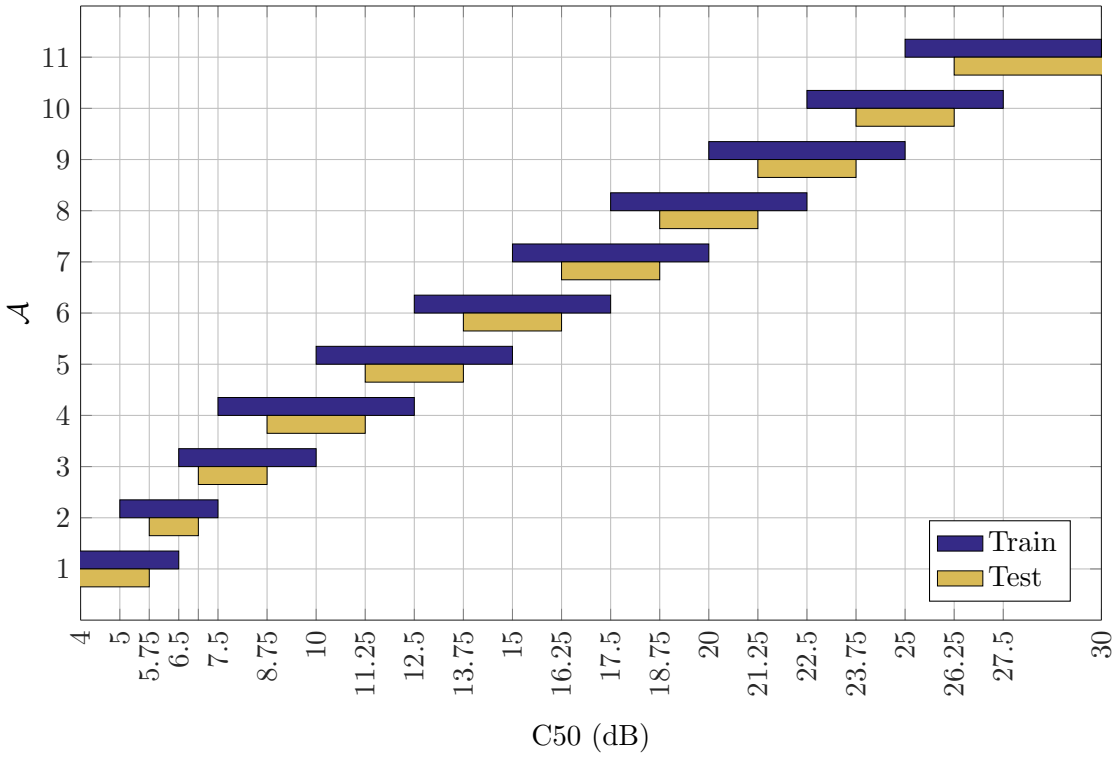

Figure 3.16: MS11 configurations to train the acoustic models (blue bars) by overlapping the training data and recognize the testing data (light brown bars) according to C_{50}.

3.4.4 Experimental setup

The methods presented are evaluated with the REVERB Challenge database [74] introduced in Section 3.3.2. The evaluation metric employed to evaluate the performance of these methods is the WER

$$\text{WER} = \frac{N_{del} + N_{ins} + N_{sub}}{N_{wrd}} \tag{3.20}$$

where N_{wrd} is the number of words in the reference, N_{del} is the number of deletions, N_{sub} is the number of substitutions and N_{ins} the number of insertions.

3.4.5 Results

Methods described in Section 3.4.3 are tested using NIRA-CART and NIRA-BLSTM to estimate C_{50} and then the performance of each method is compared in terms of the WER

obtained using the REVERB Challenge ASR task [74]. The ASR evaluation tool is based on the HTK provided by the REVERB Challenge. It uses MFCC features including Delta and Delta-Delta coefficients and tied-state HMM acoustic models with 10 Gaussian components per state for clean-condition models and 12 Gaussian components per state for multi-condition models.

Table 3.8 shows the average WER achieved with the non-reverberant recordings (Clean), simulated reverberant recordings (Sim.) and real reverberant recordings (Real) of the REVERB Challenge evaluation test set including the average of all subsets in the last column, while Table 3.9 and Table 3.10 show in more detail these results for each scenario. Moreover, Fig. 3.17 summarizes these results, displaying the average WER for development test set and evaluation test set.

In addition, Table 3.8 also includes in round brackets the WER achieved for the simulated reverberant recordings using the ground truth C_{50}. It shows that NIRA-CART tends to provide higher WER than when using the ground truth C_{50} unlike NIRA-BLSTM which tends improve the ASR performance. This result suggests that ASR performance is not entirely correlated with C_{50}, which is in accordance to Table 2.3.2.1. In fact Table 2.3.2.1 also shows that NIRA-BLSTM provides the same correlation factor compared to C_{50} ground truth while increasing the mutual information with ASR performance. This evidence might explain the lower WER achieved with NIRA-BLSTM against using ground truth C_{50}.

Baseline methods are also tested in order to compare the performance. The baseline methods consist of decoding the data using the two acoustic models provided in the REVERB Challenge: the acoustic model trained with non-reverberant data (Clean-cond.) and the acoustic model trained with reverberant data (Multi-cond.). The performance of these baselines are shown in the first two rows of Table 3.8, Table 3.9 and Table 3.10. Clean-cond. models provide a better performance in non-reverberant environments whereas Multi-cond. models provide a significant reduction in WER for reverberant environments. The difference in WER achieved in Table 3.9 is due to the fact that the sets *R1*, *R2* and *R3* are created using different utterances.

	Clean *Avg.*	**Sim.** *Avg.*	**Real** *Avg.*	**Avg.**
Clean-cond.	12.21	52.22	89.17	48.04
Multi-cond.	30.13	29.50	56.94	34.67
		NIRA-CART		
Clean&Multi cond.	13.51	29.29 (29.05)	56.94	30.02
C50FV	28.65	29.72 (29.58)	56.84	34.37
C50HLDA	25.52	27.78 (27.46)	55.00	32.12
MS3	22.17	27.22 (26.68)	54.57	30.82
MS3+C50HLDA	19.90	25.24 (24.51)	52.51	28.75
MS5	22.32	26.35 (26.23)	54.38	30.35
MS5+C50HLDA	20.07	24.80 (24.44)	52.65	28.58
MS8	21.57	26.10 (25.41)	53.17	29.80
MS8+C50HLDA	19.69	24.08 (23.50)	**51.04**	27.79
MS11	21.10	26.04 (25.83)	56.62	30.26
MS11+C50HLDA	19.83	24.24 (23.77)	53.13	28.30
MS14	21.34	25.97 (25.41)	55.13	30.02
MS14+C50HLDA	19.38	23.75 (23.57)	52.31	27.76
MS18	21.96	25.97 (25.39)	55.85	30.32
MS18+C50HLDA	20.73	23.95 (23.51)	53.12	28.38
		NIRA-BLSTM		
Clean&Multi cond.	**12.23**	29.04 (29.05)	56.94	29.54
C50FV	28.91	29.61 (29.58)	56.99	34.40
C50HLDA	25.56	27.47 (27.46)	53.42	31.68
MS3	20.67	26.84 (26.68)	55.22	30.33
MS3+C50HLDA	18.75	24.65 (24.51)	54.02	28.39
MS5	21.31	26.27 (26.23)	54.29	30.02
MS5+C50HLDA	19.42	24.44 (24.44)	52.38	28.16
MS8	19.97	25.51 (25.41)	53.71	29.13
MS8+C50HLDA	18.52	23.64 (23.50)	51.90	27.39
MS11	18.73	25.52 (25.83)	55.05	29.05
MS11+C50HLDA	17.85	23.50 (23.77)	52.51	27.24
MS14	18.95	25.27 (25.41)	54.57	28.88
MS14+C50HLDA	17.57	**23.26** (23.57)	52.48	27.03
MS18	18.38	25.13 (25.39)	55.86	28.88
MS18+C50HLDA	16.98	**23.26** (23.51)	52.68	**26.90**

Table 3.8: WER (%) averages obtained in evaluation dataset. First two rows correspond to the baseline methods and the remainder are the methods proposed in this work. Best performance results in each column are shown in bold and performance obtained with ground truth C_{50} is shown between brackets.

	Clean		
	R1	*R2*	*R3*
Clean-cond.	12.83	12.20	11.62
Multi-cond.	30.29	30.00	30.10
	NIRA-CART		
Clean&Multi cond.	13.98	13.76	12.81
C50FV	28.87	28.80	28.29
C50HLDA	25.84	24.97	25.76
MS3	22.31	21.64	22.59
MS3+C50HLDA	19.91	19.87	19.95
MS5	22.72	21.39	22.86
MS5+C50HLDA	20.18	19.57	20.47
MS8	21.94	20.69	22.11
MS8+C50HLDA	20.62	19.07	19.38
MS11	21.70	20.04	21.58
MS11+C50HLDA	20.67	19.76	19.06
MS14	21.57	20.63	21.84
MS14+C50HLDA	19.77	19.07	19.31
MS18	22.26	21.13	22.52
MS18+C50HLDA	21.47	20.31	20.41
	NIRA-BLSTM		
Clean&Multi cond.	**12.88**	**12.22**	**11.60**
C50FV	29.02	29.06	28.65
C50HLDA	25.69	24.91	26.09
MS3	20.89	20.13	21.02
MS3+C50HLDA	18.94	18.41	18.92
MS5	21.59	20.68	21.67
MS5+C50HLDA	19.21	19.17	19.89
MS8	20.35	19.41	20.17
MS8+C50HLDA	19.15	18.20	18.22
MS11	18.98	18.33	18.90
MS11+C50HLDA	18.18	18.06	17.33
MS14	19.26	18.75	18.85
MS14+C50HLDA	17.66	17.78	17.28
MS18	18.76	18.14	18.24
MS18+C50HLDA	17.37	16.95	16.64

Table 3.9: WER (%) obtained with the non-reverberant part of the evaluation dataset. First two rows correspond to the baseline methods and the remainder are the methods proposed in this work. R1, R2 and R3 represent the room number one, two and three respectively. Best performance results in each column are shown in bold.

3.4.5.1 C_{50} as a new feature

The C50FV method provides a similar performance compared to the baselines. This outcome is due to the fact that diagonal covariance matrices are used to build the acoustic

	Sim.						Real	
	R1		*R2*		*R3*		*R1*	
	near	*far*	*near*	*far*	*near*	*far*	*near*	*far*
Clean-cond.	17.91	25.67	42.85	83.70	54.22	89.08	90.19	88.15
Multi-cond.	20.60	21.09	23.70	38.72	28.08	44.86	58.45	55.44
	NIRA-CART							
Clean&Multi cond.	18.67	21.59	23.83	38.72	28.15	44.86	58.45	55.44
C50FV	20.62	20.74	23.12	39.14	28.19	46.61	58.19	55.50
C50HLDA	18.38	19.99	21.34	37.03	27.44	42.55	55.92	54.09
MS3	18.08	19.82	21.92	35.94	27.35	40.25	55.64	53.51
MS3+C50HLDA	17.16	19.40	20.60	32.67	25.37	36.32	53.53	51.49
MS5	16.32	18.52	20.49	36.34	25.85	40.62	55.35	53.41
MS5+C50HLDA	16.44	17.93	19.91	32.51	24.45	37.62	53.66	51.65
MS8	16.72	19.32	20.79	34.02	26.50	39.31	53.24	53.11
MS8+C50HLDA	15.72	18.26	19.79	**30.76**	24.16	35.85	**52.06**	**50.03**
MS11	16.50	18.99	21.14	34.75	25.85	39.09	57.87	55.37
MS11+C50HLDA	16.10	17.79	19.95	31.58	23.90	36.21	54.77	51.49
MS14	16.50	19.06	21.37	34.64	24.83	39.50	55.61	54.66
MS14+C50HLDA	15.88	17.93	19.73	30.78	22.39	35.86	52.67	51.96
MS18	16.25	19.13	21.19	34.96	24.94	39.40	56.50	55.20
MS18+C50HLDA	15.64	18.23	19.79	31.15	22.83	36.15	53.78	52.46
	NIRA-BLSTM							
Clean&Multi cond.	17.89	21.09	23.70	38.72	28.08	44.86	58.45	55.44
C50FV	20.48	20.40	23.07	39.09	27.96	46.73	58.86	55.13
C50HLDA	18.98	20.01	20.86	36.40	26.58	42.07	54.62	52.23
MS3	16.93	19.18	21.79	35.99	27.25	39.98	55.96	54.49
MS3+C50HLDA	15.93	18.35	20.00	32.51	24.89	36.26	54.81	53.24
MS5	16.06	18.43	20.47	36.28	25.99	40.44	54.84	53.75
MS5+C50HLDA	16.08	17.13	19.55	32.34	24.18	37.43	53.85	50.91
MS8	15.98	18.35	20.18	34.22	25.44	38.95	53.98	53.44
MS8+C50HLDA	15.50	16.93	19.52	30.97	23.49	35.49	52.79	51.01
MS11	15.81	18.04	20.29	34.75	25.08	39.24	55.92	54.19
MS11+C50HLDA	14.66	**16.79**	19.07	31.56	22.95	36.02	53.40	51.62
MS14	15.55	17.76	20.02	34.70	24.77	38.87	55.76	53.38
MS14+C50HLDA	14.72	17.35	**18.54**	31.16	22.47	**35.39**	53.85	51.11
MS18	15.37	17.25	19.97	34.36	24.74	39.17	56.79	54.93
MS18+C50HLDA	**14.61**	**16.79**	18.86	31.19	**22.32**	35.85	53.91	51.45

Table 3.10: WER (%) obtained with the reverberant part of the evaluation dataset. First two rows correspond to the baseline methods and the remainder are the methods proposed in this work. R1, R2 and R3 represent the room number one, two and three respectively. Best performance results in each column are shown in bold.

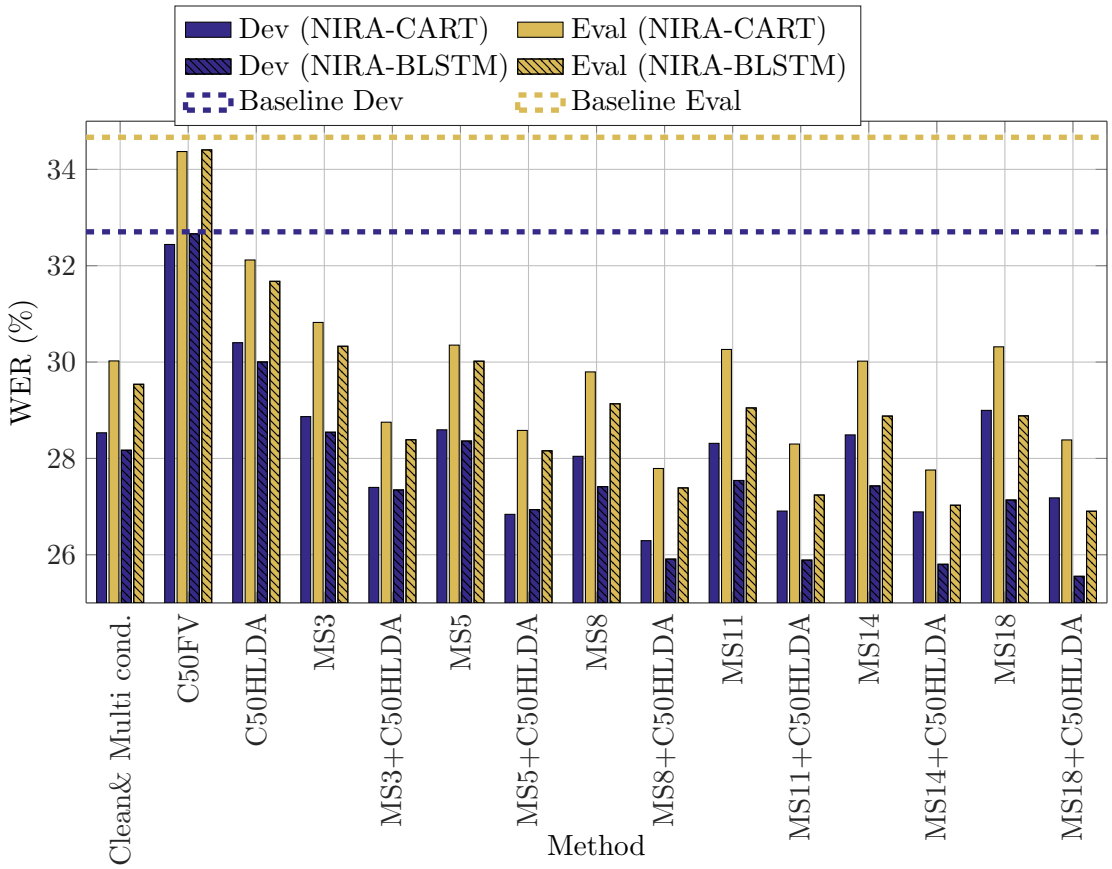

Figure 3.17: Comparison of the ASR performance of several methods (bars) against the baselines (dotted lines) for development test set (blue) and evaluation test set (light brown) using both C_{50} estimators (NIRA-CART and NIRA-BLSTM).

model. Therefore this feature only provides information regarding the probability of observing the acoustic unit in this reverberant environment not taking into account any possible dependences with the MFCCs.

On the other hand, the C50HLDA method described in Section 3.4.3.1 outperforms on average the WER obtained with the baselines. The main reason for this result is the use of the discriminative transformation matrix to combine the feature space. Regarding the C_{50} estimator employed, NIRA-BLSTM provides similar WER to that obtained with NIRA-CART for this configuration. This small performance difference suggests that C50HLDA does not strongly depend on the accuracy of the estimations. Furthermore, the averaged WER obtained by applying HLDA to the feature space without the C_{50} feature, thus reducing the dimension of the transformed spaced by 1, is 32.20%. This

result supports previous suggestion about the dependence of C_{50} estimation accuracy upon C50HLDA performance and moreover indicates that the improvement achieved with C50HLDA is mainly due to the HLDA transformation.

3.4.5.2 Model selection

Table 3.8, Table 3.9 and Table 3.10 also display the performance obtained with the methods described in Section 3.4.3.2 based on model selection. First, they show that a considerable WER reduction of the baseline is achieved by employing the two acoustic models provided by REVERB Challenge and exploiting our estimate of C_{50} to select the most appropriate model for each utterance between them (i.e., Clean&Multi cond.). Further improvement is achieved by training more reverberant models. The MS3 configuration employs three reverberant models (Fig. 3.15(a)) and the performance in reverberant conditions is improved in most of the situations but the error rate has on average been increased with respect to Clean&Multi cond. mainly due to the poor performance in clean environments. The performance of this configuration is slightly improved by overlapping the training data to build the acoustic models (MS5). Increasing the number of models trained using overlapping ranges of C_{50} (i.e., MS8, MS11, MS14 and MS18) results in further WER reductions. Table 3.8 indicates that no further improvement is on average achieved with MS18 compared to MS14, and consequently the maximum number of models investigated in this thesis is 18.

For these experiments, the best performance is obtained with MS8 using NIRA-CART C_{50} estimator (WER = 29.8%), whereas NIRA-BLSTM provides the lowest WER with MS14 (WER = 28.9%). This is due to the fact that NIRA-BLSTM achieves more accurate C_{50} estimations than NIRA-CART, hence it is able to select acoustic models trained with a narrower, and therefore better matched, C_{50} range.

3.4.5.3 Model selection including C_{50} in the feature vector

The performance of the full system presented in Fig. 3.14 is now discussed. A significant improvement is observed by combining model selection with the approach of including C_{50}

in the feature vector; the WER is decreased by approximately 2% absolute with respect to the error achieved by using only model selection. NIRA-CART offers the best performance with MS8+C50HLDA (WER = 27.8%) and NIRA-BLSTM with MS14+C50HLDA (WER = 26.9%), which outperforms the best baseline method (Multi-cond.) by 6.9% and 7.8% respectively in the evaluation set.

Table 3.8, Table 3.9 and Table 3.10 highlight in bold the lowest WER obtained in each data set. The best performance in reverberant conditions is achieved with this full system (i.e. MS$N_{\mathcal{A}}$+C50HLDA), however Clean&Multi cond. shows the best performance in non-reverberant condition. This is mainly because all the data used to train MS$N_{\mathcal{A}}$+C50HLDA is reverberant data while Clean&Multi cond. uses reverberant and clean data to train the acoustic models. Therefore MS$N_{\mathcal{A}}$+C50HLDA could be further improved by including a clean acoustic model to recognize the non-reverberant data.

Figure 3.17 shows when using a more accurate C_{50} estimator, i.e. NIRA-BLSTM against NIRA-CART, the WER is further reduced.

The method proposed in Fig. 3.14 may potentially be complementary to some other reverberation-robust speech recognition methods, such as applying speaker adaptation, acoustic model adaptation or preprocessing schemes (e.g. beamforming) [132]. For example, when performing an unsupervised acoustic model adaptation using Constrained Maximum Likelihood Linear Regression (CMLLR) with the best method proposed in this work (MS14+C50HLDA using NIRA-BLSTM) the average WER is further reduced to 24.82%, this is, a relative Word Error Rate Reduction (WERR) of 8.11% with respect to the best baseline of the REVERB Challenge using CMLLR.

3.4.6 Conclusions

Various approaches for single-channel reverberant speech recognition using *clarity index* (C_{50}) estimation have been presented. One investigated approach was to include C_{50} estimated from two different estimators (NIRA-CART and NIRA-BLSTM) as an additional feature in the ASR system and apply a dimensionality reduction technique (i.e. HLDA) to match the original feature vector dimension. This approach helped to improve the ASR

performance of the best baseline by a relative WERR of 7.35% for the NIRA-CART and NIRA-BLSTM. This improvement was shown to be in a significant part due to the HLDA transformation. Another approach was to use the C_{50} information to perform acoustic model selection, which in turn gave a relative WERR of 14.04% with NIRA-CART and 17.07% with NIRA-BLSTM. The best performance was achieved by combining both approaches and using NIRA-BLSTM, leading to a relative WERR of 22.41% (7.77% absolute WERR). It is worth noting that only data from the REVERB Challenge data sets was used to train all the models employed in the system (including the C_{50} estimator). Furthermore the method presented is complementary to other techniques such as CMLLR and an example combination was shown to improve further the best performance, increasing the relative WERR to 29.8%. A comparison of the method proposed in this section with all the method submitted to the REVERB Challenge can be found in [99].

As expected, more accurate C_{50} estimations lead to a further reduction in the final WER. Regarding the two algorithms exploited in this study, NIRA-BLSTM is more accurate than NIRA-CART by 1.6 dB RMSD, which results in relative WERR of 3.24%. These results clearly indicate that C_{50} can be successfully used for reverberant speech recognition tasks and the accuracy in the C_{50} estimation is crucial.

Chapter 4

Speaker diarization based on spatial features

In this chapter, two methods to perform speaker diarization from the input speech signal are presented. Firstly, in Section 4.2 a single-channel approach is described based on MFCC features and DRR estimation. Secondly, Section 4.3 introduces a multi-channel approach to statistically model the Time Delay of Arrival (TDOA) estimates obtained using Generalized Cross Correlation with Phase Transform (GCC-PHAT) algorithm on pairs of microphones for the task of robust diarization in reverberant environments. Both methods are evaluated in the context of multi-talker meeting scenarios recorded using distant microphones. Unless otherwise stated, it is assumed throughout this chapter that the number of speakers in the recordings is known.

The research presented in this chapter relates in part to the following publications [16] [20]. The contribution of the thesis in [16] is to provide an estimation of the DRR, adapt NIRA framework for the database employed in the experiments and evaluate its performance.

4.1 Introduction

Speaker diarization systems have gained much importance over the past five years in overcoming key challenges faced by automatic meeting transcription systems. These systems aim at segmenting the audio signal into homogeneous sections with only one active speaker and answer the question "who spoke when?". In this chapter the term diarization refers to the process of seeking fragments of audio which correspond to the same speaker regardless the speaker's identity.

Figure 4.1 shows the waveform of a recording with two speakers. In this case, the diarization system seeks first for segments in the audio with speaker activity. Following this segmentation process, a clustering process is applied to find the segments where the same speaker is active and thus associate these segments with the same speaker index. The perfect diarization result is illustrated in Fig. 4.2.

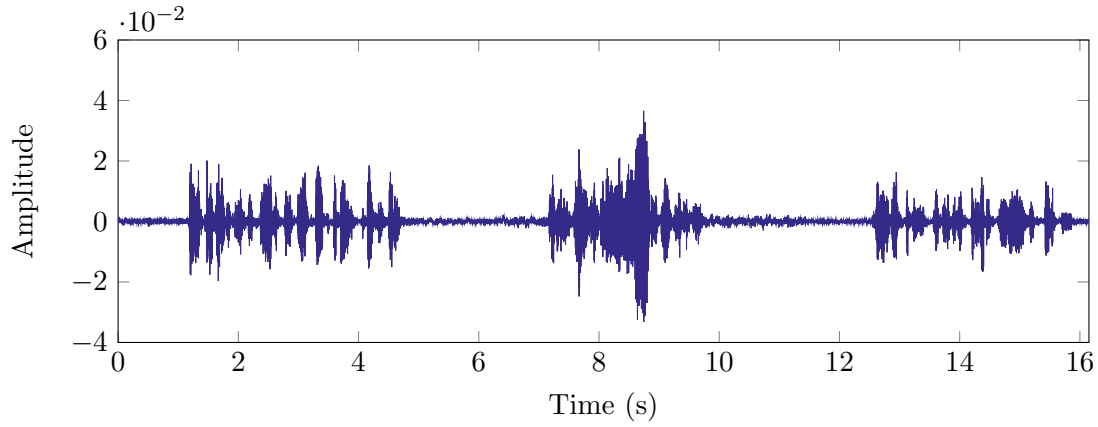

Figure 4.1: Recording example without diarize.

Speaker diarization provides important information in multiple applications such as speaker indexing or rich transcription of multi-speaker audio streams. Furthermore, this information can be used to improve the performance of ASR systems by allowing effective speaker acoustic model adaptation. Input audio streams may be generated in multiple scenarios such as call centers or meetings. Approaches presented in this chapter are focused on latter scenarios. Figure 4.3 illustrates a meeting scenario with two speakers and two microphones located in a room. Usually, the positions of either the speaker or

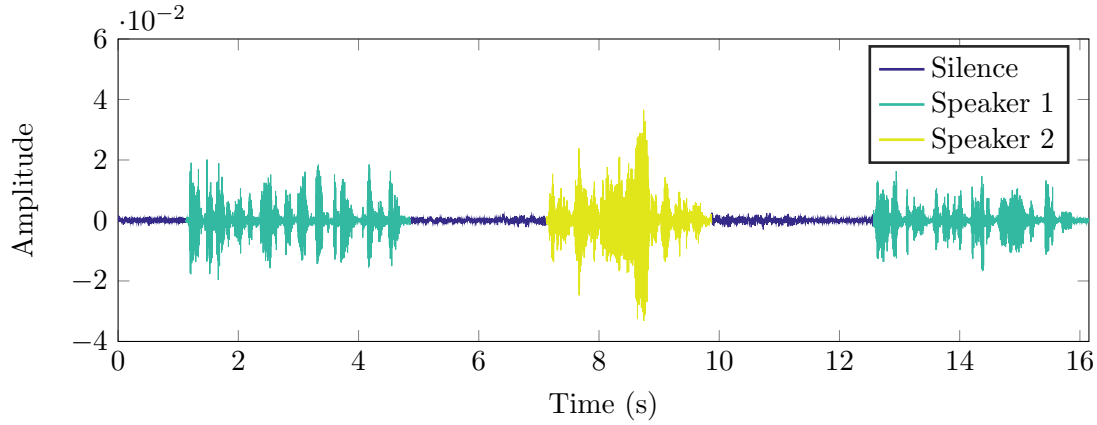

Figure 4.2: Recording example with perfect diarization.

the microphones are unknown. Additionally, the recordings can be distorted by noise, reverberation or non-speech acoustic events (e.g. music) thus degrading the diarization performance [7].

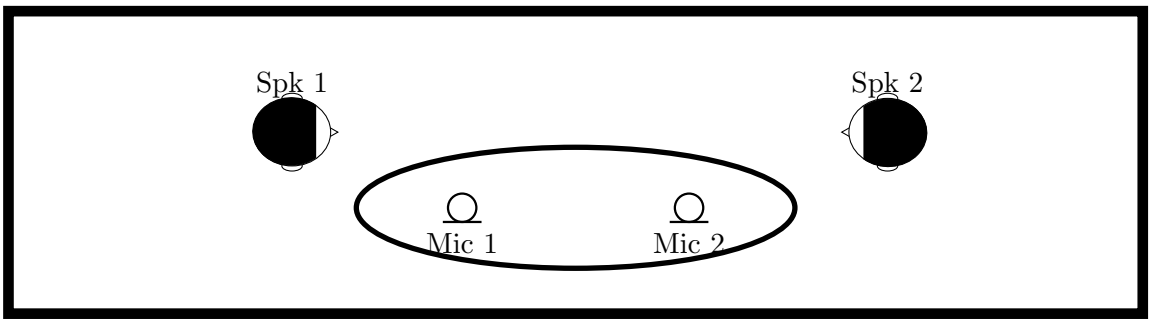

Figure 4.3: Meeting scenario in a room with two speakers, i.e. Spk1 and Spk2, located close to a table where there are two microphones.

Current state-of-the-art algorithms can only utilize spatial information when multi-microphone recordings are available. This information is usually related to the TDOA [133] which represents the time delay of the same signal in two different microphones, or based on steered response power method [134] that seeks the location where the beamformer created with all microphones provides the maximum power output. In single-microphone scenarios this feature is infeasible to compute and therefore commonly speech features as MFCC or Perceptual Linear Predictive (PLP) are typically used to perform diarization. In contrast, the approach presented in this chapter at Section 4.2 is able to leverage spatial

information by estimating this information from single-channel recordings.

4.1.1 Background and literature review

State-of-the-art diarization approaches [7] fall into two main categories: bottom-up and top-down. The former is initialized for the entire audio input with many clusters (typically more than the expected number of speakers), where a cluster refers to a collection of features corresponding to temporal segments of the speech signal, which are merged successively until only one speaker per cluster remains, while the latter starts with only one cluster and adds new clusters until all speakers are correctly modelled. Figure 4.4 represents the general architecture of state-of-the-art diarization system. Feature extraction, cluster initialization, split/merging procedure or stop criterion are important issues in these systems for which different solutions have been proposed in the literature [7, 135].

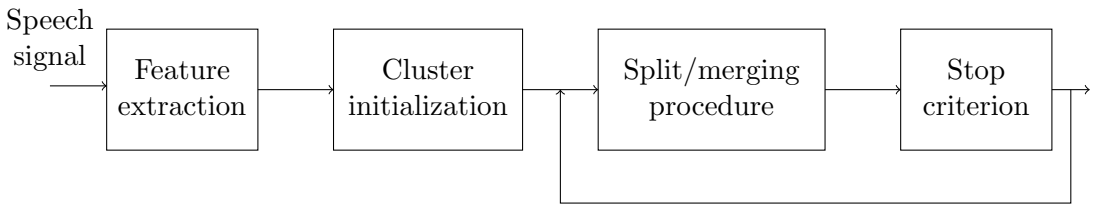

Figure 4.4: Generalized diarization block diagram.

State-of-the-art diarization methods can also be broken down into two main groups depending on the microphone configuration: single-channel and multi-channel approaches.

Single-channel speaker diarization algorithms generally discriminate different speakers using speech dependent features such as MFCC or PLP coefficients [136] commonly extracted from data captured by a close talking microphone [137]. In recent years, Log Mel-filterbanks are employed in DNN-based systems [138] or i-vector features widely used in speaker recognition [139].

When multi-channel signals are available, TDOA estimates are frequently used to perform diarization. In [133], a framework to combine these TDOAs with MFCCs is proposed. In [140] the diarization is performed using the TDOAs obtained from all possible combinations of microphones. An unsupervised discriminant analysis method,

an LDA-like formulation without the need of speaker labels, is then applied to these TDOAs to transform the input space into a new feature space. These new features are then used to diarize using a standard agglomerative clustering approach. The diarization system in [141] is based on estimates of the phoneme, vowel and consonant classes, which are extracted from a phoneme recognizer. Speaker change points and speaker clusters are calculated using the Bayesian Information Criterion (BIC) [142]. This criterion is computed from Gaussian models fitted to MFCC features computed from two successive speech segments, always using different models for each segment and for each phoneme class. A real-time meeting analyzer is presented in [143]. Several blocks of the full system are presented (e.g. dereverberation, source separation, speech recognition) along with speaker diarization which is based on clustering the Direction of arrival (DOA). Speaker diarization decisions are extracted by averaging the per frame diarization decisions over the word length. A front-end for speaker diarization based on beamforming is presented in [144]. The beamforming uses TDOAs which are computed from GCC-PHAT [145] and then post-processed by a dual pass Viterbi decoding. The first pass selects the most probable paths from N-best lists of TDOAs computed from a pair of microphones, while the second pass finds the best path given all combinations of paths between each pair of microphones computed in the first pass.

While speech features are commonly used in diarization systems, visual cues can also be included into the system to improve the final diarization performance [146]. However, this is excluded from the scope of this thesis.

4.2 Single-channel diarization enhanced with DRR estimates

In many meeting scenarios, such as teleconferencing, there is only a single microphone signal available. In such scenarios, the current state-of-the-art speaker diarization systems are unable to benefit from any spatial information as beamforming preprocessing or TDOA features can only be used when there are at least 2 microphone array signals available.

Whereas in previous single-channel speaker diarization methods, reverberation would be considered as a distortion in the signal, in this section this distortion is turned into an advantage. The DRR [22] is known to be strongly correlated with the distance between a microphone and the sound source [147].

The DRR parameter is estimated from the single-channel signal using the non-intrusive algorithm described in Section 2.3. These estimates are used as an additional feature to characterize the acoustic channel from each speaker to the microphone. The speakers are assumed to be stationary and the received signal at the microphone $y(n)$ at time instant n is given by

$$x_i(n) = \sum_{m=0}^{M-1} h_i(m) s_i(n-m), \tag{4.1}$$

$$y(n) = \sum_{i=1}^{N_{spk}} x_i(n) + \nu(n) \tag{4.2}$$

where i represents a speaker index, $h_i(m)$ is the M sample time-invariant RIR describing the acoustic propagation from the i-th speaker to the microphone, N_{spk} the total number of speakers and $\nu(n)$ is the additive noise at the microphone.

4.2.1 Baseline system

The baseline, selected for comparison in this section, is the DiarTK system proposed in [148]. It discriminates different speakers based on input feature streams, e.g. MFCC features, and relies on the Information Bottleneck (IB) principle [149]. This IB based diarization system clusters a given uniform linear segmentation $\mathcal{Y}$ of the recorded signal $y(n)$ into a set $\mathcal{C}$ of clusters, which compresses the input variable reducing $I(\mathcal{Y};\mathcal{C})$ while preserving the mutual information about a set $\mathcal{B}$ of relevance variables $I(\mathcal{C};\mathcal{B})$. This is achieved by the minimization of the following objective function using the agglomerative IB [150]:

$$I(\mathcal{Y};\mathcal{C}) - \beta_t I(\mathcal{C};\mathcal{B}) \tag{4.3}$$

where I is the mutual information and β_t is a trade-off parameter set to 0.1 according to [151]. The relevance variables correspond to the components of a background GMM trained on the entire recording. Each component of this background GMM is estimated for each segment.

At each step of the agglomerative IB, two clusters are merged so that the loss of mutual information about the relevance variables is minimum. The optimal number of clusters is determined by a threshold on the normalized mutual information $\frac{I(\mathcal{B},\mathcal{C})}{I(\mathcal{Y},\mathcal{B})}$ [133].

At the output of the agglomerative IB algorithm, the clusters are aligned with the boundaries of the initial segments. These boundaries are realigned by computing the sequence of clusters that minimizes the cost function based on the posterior distribution of the relevance variables given the input feature vector.

The baseline configuration employed in this section uses an input feature vector with 19 MFCCs extracted from the STFT of the recorded signal $y(n)$ after applying a 20 ms Hamming window with 50% overlap.

4.2.2 Proposed system

The proposed diarization system, a block diagram of which is shown in Fig. 4.5, is built on top of the system described in [148]. From the received signal $y(n)$, 2 streams of features, namely MFCC and DRR features, are extracted independently before being combined and clustered so that a label is assigned to the lth frame.

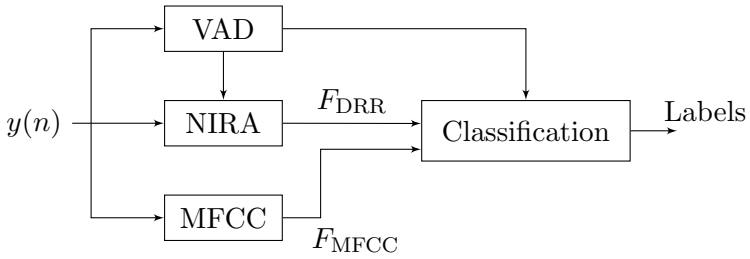

Figure 4.5: Block diagram of the proposed speaker diarization system.

The estimation of DRR is proposed to be used as a spatial feature for speaker

diarization, defined for speaker i as follows [21]

$$\mathrm{DRR}_i = 10\log_{10}\left(\frac{\sum_{m=n_d-N_w}^{m=n_d+N_w} h_i^2(m)}{\sum_{m=0}^{m=n_d-N_w} h_i^2(m) + \sum_{m=n_d+N_w}^{\infty} h_i^2(m)}\right)\mathrm{dB}, \tag{4.4}$$

where N_w is the number of samples in a rectangular window of 8 ms and n_d is the time index (in samples) of the direct path arrival in the RIR, $h_i(m)$.

To evaluate this measure of reverberation, the RIR needs to be known or estimated. However, in this work DRR is estimated non-intrusively from the reverberant signal, i.e. without information on the RIR or the source signal. The NIRA method employed is NIRA_β described in Section 2.6.2. This model is trained with recordings that contain the same DRR value for all frames within an utterance. Therefore, the estimation of DRR per frame tends to be in a reduced DRR range. In order to avoid this issue when there are different ground truth DRRs in a recording, as in this experiment, the estimations are carried out with a reduced number of frames using a rectangular window of 3 s with a 50% overlap. The size of this window is a trade-off between the minimum amount of data needed to achieve an accurate DRR estimation and the maximum period of time where the DRR is fixed.

4.2.2.1 VAD

A VAD based on the P.56 method [56, 152] is applied to extract segments with active speech from the input signal. The time boundaries of detected speech are also given to the clustering block to discard speech features that were extracted from detected silence frames. This VAD is the same as the VAD used in NIRA (Section 2.3) to detect non-speech frames.

4.2.2.2 Feature combination

A background GMM, $\boldsymbol{\theta}_{\mathcal{J}}$, is now estimated for each stream of features, $F_{\mathcal{J}}$. The VAD described in Section 4.2.2.1 is applied to exclude features extracted from estimated pauses in the training of the GMM. The combined distribution, which is required in the IB based diarization system, is then calculated as:

$$p(\mathfrak{b}|\mathfrak{y}) = \sum_{\mathcal{J}\in\{\text{MFCC, DRR}\}} p(\mathfrak{b}|\boldsymbol{\theta}_{\mathcal{J}}, \mathfrak{y})\mathcal{W}_{\mathcal{J}}$$

where $\mathfrak{b} \in \mathcal{B}$ is a relevance variable, $\mathfrak{y} \in \mathcal{Y}$ is an input feature and the stream weights, $\mathcal{W}_{\mathcal{J}}$, satisfy $\mathcal{W}_{\text{MFCC}} + \mathcal{W}_{\text{DRR}} = 1$. These weights are empirically estimated from a development set to maximize performance.

4.2.3 Experimental setup

This section describes the experimental setup used to evaluate the diarization systems. NIRA is trained on the REVERB Challenge training data. The measured RIRs used in the training and the simulated meeting data are captured in different rooms [74].

4.2.3.1 Simulated meeting data

The simulated meetings are generated by convolving clean speech with measured RIRs taken from the evaluation set of the REVERB Challenge database [74]. Recorded fan noise is added at 20 dB SNR.

4.2.3.1.1 Speech data

The nearly-anechoic speech data consists of utterances taken from the WSJCAM0 corpus [131] which are recorded by a close-talking noise-cancelling head-mounted microphone. There are, in total, 14 speakers: 7 male and 7 female. Each utterance is only used once across the simulated meetings.

4.2.3.1.2 RIRs

The RIRs of the REVERB Challenge are captured in 3 rooms of different size by a circular microphone array. For each room, the RIR that is recorded at a distance of 0.5 m is labeled as *near* and the RIR that is recorded at a distance of 2 m is labeled as *far*. The ground truth DRRs together with the reverberation time T_{60} are given in Table 4.1.

	Room 1	**Room 2**	**Room 3**
T_{60} (s)	0.25	0.5	0.7
Near DRR (dB)	16	11	11
far DRR (dB)	6	0	2

Table 4.1: T_{60} in s and DRR in dB for the *near* and *far* positions in each of the three rooms.

In each simulated meeting, the reverberant signal consists of 10 to 14 utterances spoken by 2 different speakers speaking turn-taking. Spatial differences between the 2 speakers are simulated by convolving each speakers' signal with different RIRs, i.e. the utterances spoken by one speaker are convolved with the *near* RIR while the utterances spoken by the other speaker are convolved with the *far* RIR.

In total, 42 simulated meeting audio streams are generated, 14 streams per room. Among the 42 audio streams, 18 streams contained speakers of different gender, 12 streams contained only male speakers and 12 streams contained only female speakers.

4.2.3.1.3 Noise

Fan noise signals, recorded in each room, are added to the simulated meeting data at a mean SNR of 20 dB, defined as [74]

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\sum_{n=1}^{N_{sam}}\sum_{i=1}^{N_{spk}}\tilde{x}_i^2(n)}{\sum_{n=1}^{N_{sam}}\nu^2(n)}\right), \tag{4.5}$$

$$\tilde{x}_i(n) = \tilde{h}_i * s_i(n), \tag{4.6}$$

where N_{sam} is the total number of samples in the considered recording, $\tilde{h}_i = [h_i(0), h_i(1), \ldots, h_i(n_{50}-1)]$ is the truncated RIR containing the n_{50} taps within the first 50 ms and $\tilde{x}_i(n)$ is the direct sound together with the early reflections up to 50 ms.

4.2.3.1.4 Development and evaluation sets

The simulated data is broken down into two non-overlapping subsets: a development and an evaluation set.

The development set is used to determine the optimum weights $\mathcal{W}_{\text{MFCC}}$ and $\mathcal{W}_{\text{DRR}} = 1 - \mathcal{W}_{\text{MFCC}}$ to use in the IB based diarization system to minimize the Diarization Error Rate (DER) defined below. The optimum weights are then used on the evaluation set.

4.2.3.2 Evaluation

The performance of the diarization system is commonly evaluated in terms of DER [153]. The DER corresponds to the percentage of time where the correct speaker is not detected. The DER is computed as sum of three contributions [153]:

- the percentage of time the speakers are unlabelled in speech segments (missed speaker),
- the percentage of time the speakers are estimated in silence segments (speaker false alarm),
- the percentage of time the speaker is labelled as another speaker (speaker error), which includes overlapping speakers.

The DER is computed as follows [153]:

$$\text{DER} = \frac{\text{missed speaker} + \text{speaker false alarm} + \text{speaker error}}{\text{total reference speech time}} \tag{4.7}$$

A non-scoring collar of 0.25 s is used in the evaluation to avoid small inconsistencies at the start and end of the time labels.

4.2.4 Results

In this section an analysis to find the optimal weight $\mathcal{W}_{\text{DRR}}$ used to combine DRR with MFCCs features is first presented. Then the baseline is compared with the proposed

system in terms of speaker error magnitudes. Missed speaker and speaker false alarm are excluded from this analysis because the same VAD, based on P.56 [154], is used in both cases thus these errors are exactly the same.

4.2.4.1 Development set

Figure 4.6 shows the variation of speaker error with $\mathcal{W}_{\text{DRR}}$ on the development set. It can be observed that the use of DRR features alone in the IB based diarization system does not achieve low speaker errors. However, by combining them with MFCCs, the performance of the diarization system can be improved to outperform the similar system solely relying on MFCCs.

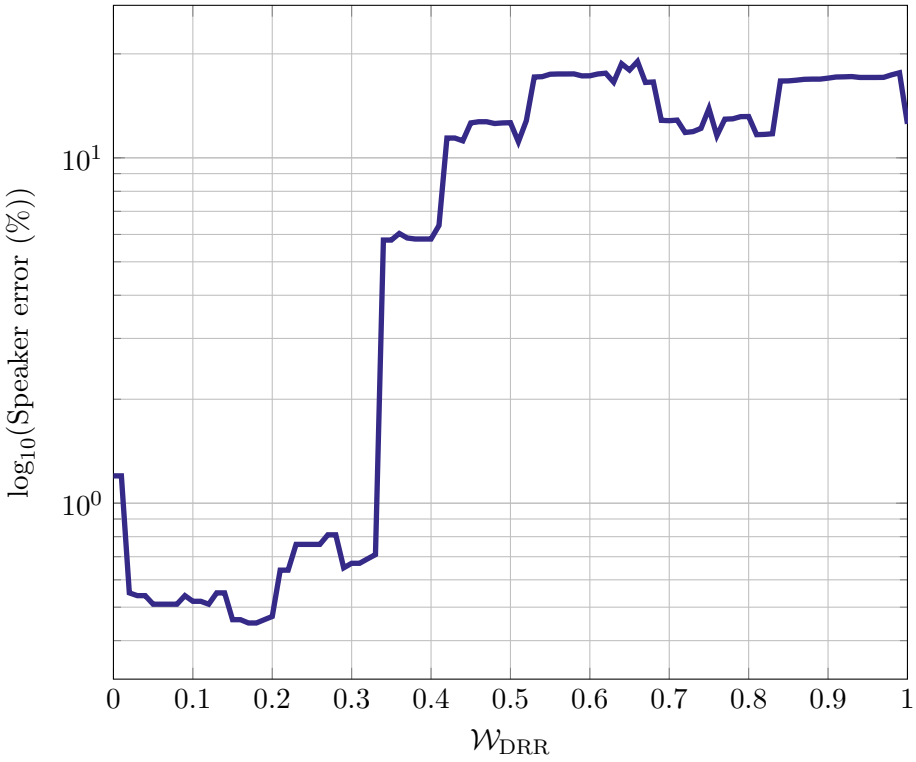

Figure 4.6: Speaker error time of the development set as a function of DRR weight ($\mathcal{W}_{\text{DRR}} = 1 - \mathcal{W}_{\text{MFCC}}$).

The lowest error in the development set is achieved for $\mathcal{W}_{\text{DRR}} = 0.18$. This weight reduces the speaker error from 1.95% to 1.2%, achieving a relative improvement of 38%.

Missed speaker and speaker false alarm percentages in this set are 0.11% and 0.45%

respectively for both methods. As shown in Table 4.2, the inclusion of DRR features in the diarization system reduces the speaker error time on average. Although the baseline and the proposed systems perform similarly in Room 1, significant improvements are observed in Room 2 and Room 3, i.e. rooms with higher amount of reverberation. A relative improvement of 79% and 41% are respectively seen in the latter 2 rooms.

	Overall	**Room 1**	**Room 2**	**Room 3**
Baseline	1.4%	0.60%	2.00%	1.60%
Proposed	0.65%	0.59%	0.42%	0.94%

Table 4.2: Mean speaker error time of the baseline and proposed method for the development set.

4.2.4.2 Evaluation set

The weights $\mathcal{W}_{\text{MFCC}} = 0.82$ and $\mathcal{W}_{\text{DRR}} = 0.18$ determined from the development set are now applied to evaluate the performance of the diarization system on the evaluation set. The missed speaker and speaker false alarm percentages are 0.32% and 0.48% respectively for both methods compared.

Figure 4.7 shows that the inclusion of DRR features provides a 34% relative reduction of the speaker error time. This error is decreased in Room 2 and Room 3 while it is increased in Room 1 by 15% on average. This degradation in performance is due to the limitations of the NIRA estimator, which was shown in Fig. 2.15 to have high estimation errors for environments with low reverberation. Analyzing the RMSD of the estimated DRR in each room, Table 4.3, the RMSD indeed decreased as T_{60} increased. The RMSD values were respectively 2.7 dB, 2.5 dB and 2.2 dB for Room 1, Room 2 and Room 3.

	Overall	**Room 1**	**Room 2**	**Room 3**
RMSD (dB)	2.5	2.7	2.5	2.2

Table 4.3: RMSD of the estimated DRR on the evaluation set.

Figure 4.8 shows that estimated DRRs mostly take values around 10 dB and 2 dB, depending on the identity of the active speaker. This figure corresponds to the simulated meeting which had the highest speaker error rate, i.e. 13.7%. The presence of the DRR

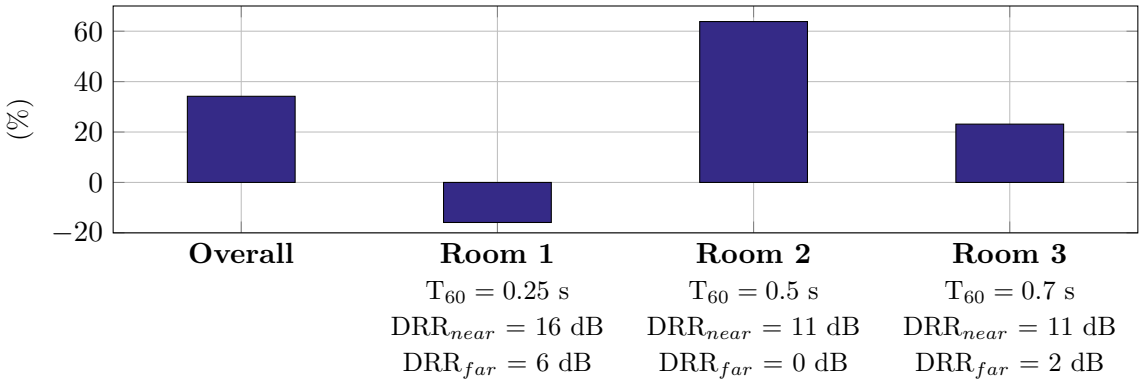

Figure 4.7: Relative improvement in speaker time error by inclusion of DRR features.

features decreases the speaker error to 4.1%, achieving a relative improvement of 70%.

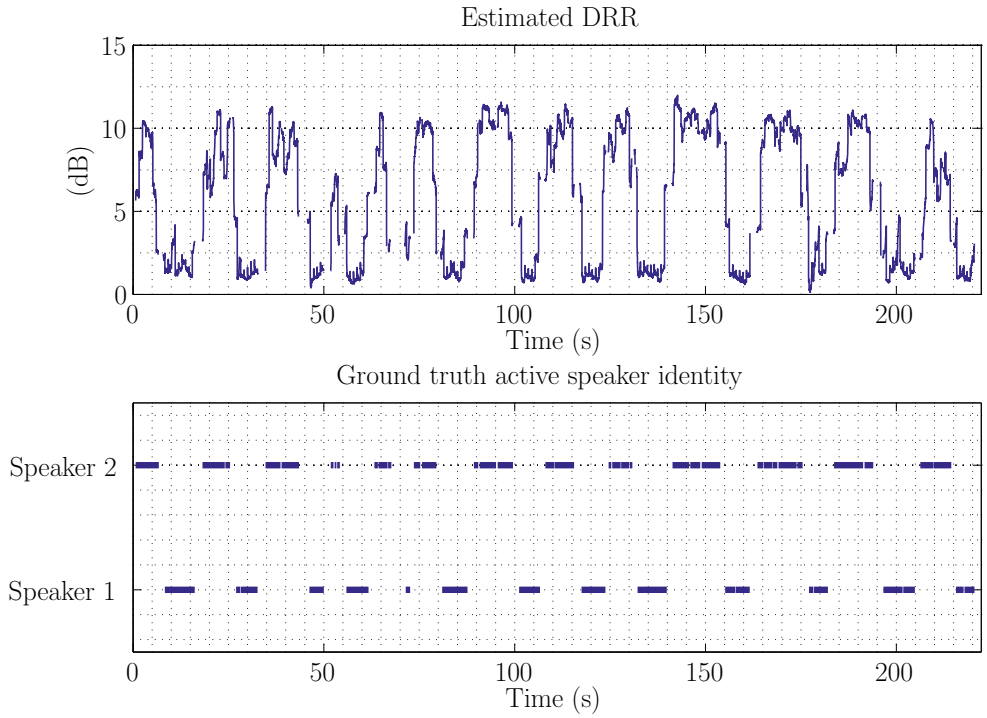

Figure 4.8: Estimated DRR along with the ground truth speaker identity.

Table 4.4 shows that the proposed system decreases the mean speaker error time when the speakers have the same gender while the mean speaker error time slightly increases when the speakers have different gender. This shows that the DRR feature is beneficial when the MFCCs have low discrimination capabilities.

	Male - Male	Mixed	Female - Female
Baseline	1.07%	0.76%	3.20%
Proposed	0.74%	0.77%	1.68%

Table 4.4: Mean speaker error time broken down by gender for the evaluation set.

4.2.5 Conclusions

A method that takes advantage of the presence of the speaker-dependent variation of reverberation has been explored in single-channel recorded meetings for speaker diarization problems. The DRR has been shown to be a feature that can discriminate different speakers located in a room at different positions relative to the microphone.

By combining non-intrusive estimates of the DRRs with MFCC features, the proposed diarization system has been evaluated on simulated meeting data created using recorded RIRs and noise signals and has been shown to give a relative improvement of 34% on average in terms of speaker error time. The standard deviation of the speaker error time has also been reduced in the 2 rooms with higher T_{60}.

4.3 Multi-channel diarization based on robust TDOA modelling

In a multi-channel diarization system the received signal $y_p(n)$ at the pth microphone at time index n is given by

$$x_{i,p}(n) = \sum_{m=0}^{M-1} h_{i,p}(m) s_i(n-m), \tag{4.8}$$

$$y_p(n) = \sum_{i=1}^{N_{spk}} x_{i,p}(n) + \nu_p(n) \tag{4.9}$$

where $x_{i,p}(n)$ is the contribution of the ith speaker at the pth microphone, $h_{i,p}(m)$ is the room impulse response between the ith speaker and the pth microphone and $\nu_p(n)$ is the additive noise present at pth microphone.

The TDOA is a common feature extracted in multi-microphones environments.

This parameter represents the difference of arrival times when a signal originating from a point source is recorded by microphones at two different positions relative to the source. Figure 4.9 illustrates the TDOA obtained from two microphones and two different speakers. Selecting the microphone 1 as the reference signal, the TDOA of speaker 1 (TDOA_{spk1}) is positive, however the TDOA of speaker 2 (TDOA_{spk2}) is negative since speaker 2 is closer to microphone 2 and therefore the signal arrives first to this microphone and then to microphone 1. This fact indicates that TDOA can potentially be associated with a certain speaker to perform diarization.

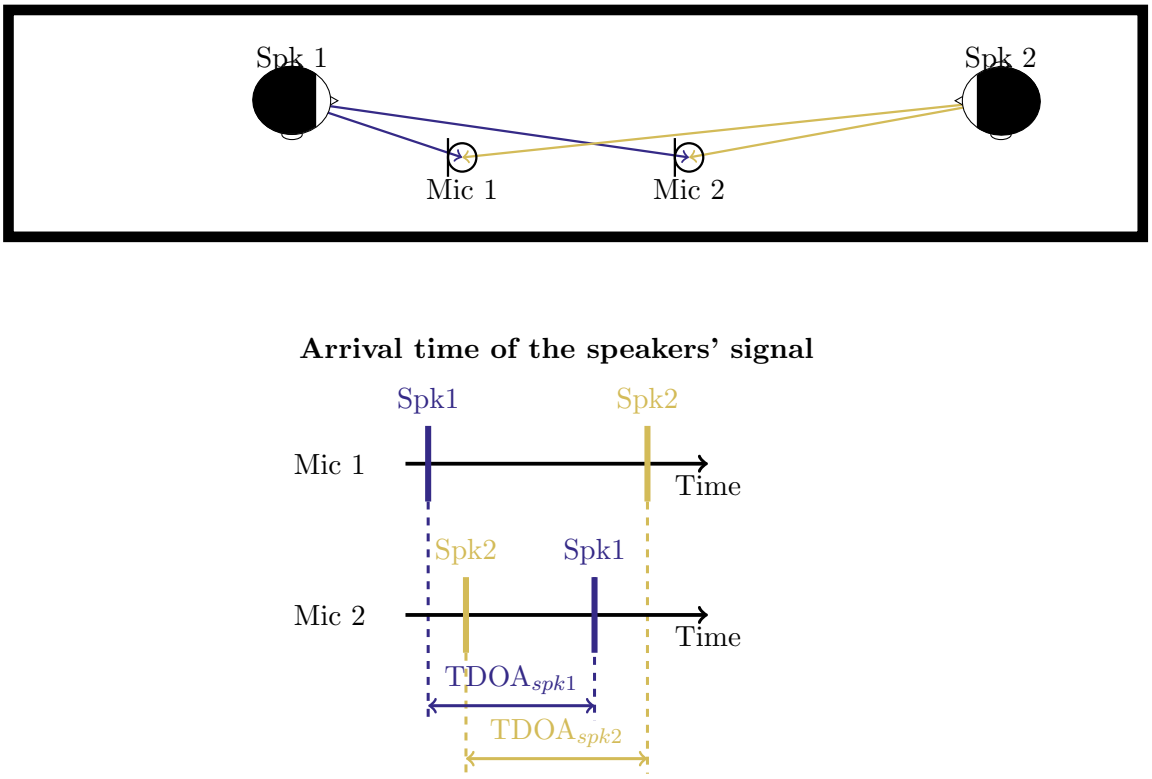

Figure 4.9: Illustration of the TDOA concept. Assuming Mic 1 is used as a reference, TDOA_{spk1} is positive and TDOA_{spk2} is similar to TDOA_{spk1} in magnitude but negative.

4.3.1 Baseline system

The baseline, referred as the Optimal Geometry baseline (OG), is based on the fact that one of the microphones used to compute the TDOAs is closer to one speaker and the other microphone is closer to the other speaker. Therefore, under this assumption, pos-

itive TDOAs are obtained when one of the speakers is talking and negative TDOAs are obtained when the sound is coming from the other speaker. This assumption is valid for this simulated scenario, however in most of the real scenarios the position of the microphones is unknown. The baseline used for these real scenarios is the system described in Section 4.2.1. For comparison purposes with the method proposed in the following section, the input features of this baseline are the same TDOAs features used in our system.

4.3.2 Proposed system

In this section, the TDOAs are modelled in a robust manner so that speaker diarization is more accurately performed. The TDOAs are estimated using GCC-PHAT [145] which computes the normalized cross-correlation between two signals in the frequency-domain

$$\mathrm{G}_{\mathrm{PHAT}}(f) = \frac{Y_1(f) \cdot Y_2(f)^*}{|Y_1(f) \cdot Y_2(f)^*|}, \tag{4.10}$$

where $Y_1(f)$ and $Y_2(f)$ are the Fourier transforms of two input signals. The TDOA τ_l for the frame l is found by maximizing $\mathrm{R}_{\mathrm{PHAT}}(\tau)$,

$$\tau_l = \arg\max_{\tau} \mathrm{R}_{\mathrm{PHAT}}(\tau), \tag{4.11}$$

where $\mathrm{R}_{\mathrm{PHAT}}(\tau)$ is the inverse Fourier transform of (4.10).

The frame size employed to compute Fourier transforms $Y_1(f)$ and $Y_2(f)$ is selected as a balance between the robustness required in the cross-correlation estimation and the time resolution of changes needed in TDOA. A frame size of 500 ms with a 87.5% of overlap between consecutive frames is used in the current work because this was found to give a good balance of these factors in our experiments.

The total number of different TDOA streams J, referred as channels in this section from this point onwards, feasible to compute from an N_{mic} microphone setup is given by the following expression,

$$J = \frac{N_{mic} \cdot (N_{mic} - 1)}{2}. \tag{4.12}$$

Each of the J channels comprise a total of N_{TDOA}.

The aim is to find, for each frame l, the speaker index i that maximizes the posterior probability of the speaker model $\boldsymbol{\theta}_i$ given the TDOA sample τ_l

$$\arg\max_i P(\boldsymbol{\theta}_i|\tau_l), \tag{4.13}$$

$$P(\boldsymbol{\theta}_i|\tau_l) = \frac{P(\tau_l|\boldsymbol{\theta}_i)\cdot P(\boldsymbol{\theta}_i)}{\sum_{e=1}^{N_{spk}} P(\tau_l|\boldsymbol{\theta}_e)\cdot P(\boldsymbol{\theta}_e)}. \tag{4.14}$$

The denominator of (4.14) is independent of i and hence it can be omitted from the maximization, thus the final maximization expression is

$$\arg\max_i P(\tau_l|\boldsymbol{\theta}_i)\cdot P(\boldsymbol{\theta}_i). \tag{4.15}$$

Figure 4.10 shows the block diagram of the method for a setup with N_{mic} microphones. The *modelling* block in Fig. 4.10 is described in Section 4.3.2.1 and Section 4.3.2.2. *Alignment within channel* and *alignment between channels* are presented in Section 4.3.2.3. Finally, Section 4.3.2.4, Section 4.3.2.5 and Section 4.3.2.6 introduce the *decoding* block.

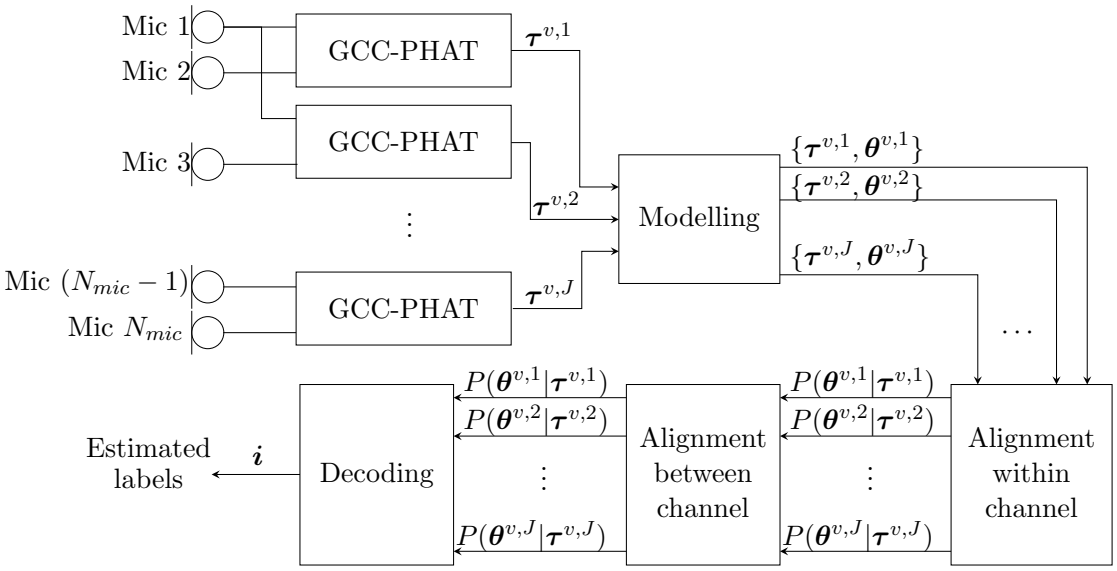

Figure 4.10: Block diagram of the method. The symbol v indicates the local modelling window index introduced in Section 4.3.2.2.

4.3.2.1 Computation of the speaker model

A GMM is represented by $\boldsymbol{\theta} = (\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\sigma})$ and can be parametrized by the a priori vector ($\boldsymbol{\lambda}$), the mean vector ($\boldsymbol{\mu}$) and the covariance matrix ($\boldsymbol{\sigma}$). The parameters of the individual mixtures are represented by $\boldsymbol{\theta}_i = (\lambda_i,\mu_i,\sigma_i)$.

A total of $N_{spk} + 1$ mixtures are considered in this approach, i.e. $\boldsymbol{\theta} = (\boldsymbol{\theta}_B,\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\cdots,\boldsymbol{\theta}_{N_{spk}})$, N_{spk} mixtures to model the speakers' TDOAs and an additional mixture to model the noisy estimates

$$\begin{aligned} \text{Background noise model}: \quad \boldsymbol{\theta}_B &= (\lambda_B, \mu_B, \sigma_B). \\ \text{Speaker 1 model}: \quad \boldsymbol{\theta}_1 &= (\lambda_1, \mu_1, \sigma_1). \\ \text{Speaker 2 model}: \quad \boldsymbol{\theta}_2 &= (\lambda_2, \mu_2, \sigma_2). \\ . \quad . \quad . \\ \text{Speaker } N_{spk} \text{ model}: \quad \boldsymbol{\theta}_{N_{spk}} &= \left(\lambda_{N_{spk}}, \mu_{N_{spk}}, \sigma_{N_{spk}}\right). \end{aligned} \tag{4.16}$$

The MLE [155] of the model parameters given the data (i.e. TDOAs) can be used to obtain $\boldsymbol{\theta}$

$$\arg\max_{\boldsymbol{\theta}} \log\ p(\boldsymbol{\tau}|\boldsymbol{\theta}), \tag{4.17}$$

where $\boldsymbol{\tau} = (\tau_1, \cdots, \tau_{N_{TDOA}})$.

In common applications, $\boldsymbol{\tau}$ can be inaccurate due to spurious noise, overlapping speakers, non-speech acoustic events or reverberation. Thus, $\boldsymbol{\theta}$ needs to be estimated robustly to these outliers.

In order to robustly estimate these model parameters $\boldsymbol{\theta}$, linear constraints are applied on the mean and the standard deviation in the EM algorithm.

4.3.2.1.1 Linear constraints on the mean

Linear constraints on the mean are determined a priori by the matrix $\mathcal{M}$ and the vector $\boldsymbol{C}$. These are defined such that the mean of the noise model μ_B is independent of the speakers' means. Additionally, the speakers' means are separated by a constant to avoid them being extremely close to each other, i.e. $\mu_1 \approx \mu_2$. Therefore, the linear constraints

on the means are

$$\boldsymbol{\mu} = \mathcal{M}\boldsymbol{\beta} + \boldsymbol{C}, \tag{4.18}$$

which can be written as

$$\begin{bmatrix} \mu_B \\ \mu_1 \\ \mu_2 \\ . \\ . \\ . \\ \mu_{N_{spk}} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ . & . \\ . & . \\ . & . \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ C_2 \\ . \\ . \\ . \\ C_{N_{spk}} \end{bmatrix}, \tag{4.19}$$

$$\begin{aligned} \mu_B &= \beta_1, \\ \mu_1 &= \beta_2, \\ \mu_2 &= \beta_2 + C_2, \\ . \quad &. \quad . \\ \mu_{N_{spk}} &= \beta_2 + C_{N_{spk}}. \end{aligned} \tag{4.20}$$

The term $C_{N_{spk}}$ is computed as the difference of the highest peak τ_{max1} and the N_{spk}-th highest peak $\tau_{maxN_{spk}}$ of the density estimation $p(\tau)$ computed from $\boldsymbol{\tau}$ using a Gaussian kernel $\mathfrak{K}(\frac{\|\tau-\tau_l\|}{\sigma})$,

$$C_{N_{spk}} = \tau_{maxN_{spk}} - \tau_{max1}, \tag{4.21}$$

$$\tau_{max1} = \arg\max_{\tau} \left\{ p(\tau) \mid \frac{dp(\tau)}{d\tau} = 0 \right\}, \tag{4.22}$$

$$\tau_{maxN_{spk}} = \arg\max_{\tau} \left\{ p(\tau) \mid \frac{dp(\tau)}{d\tau} = 0 \text{ and } \tau \neq \{\tau_{max1}, \tau_{max2}, \cdots, \tau_{max(N_{spk}-1)}\} \right\}, \tag{4.23}$$

$$p(\tau) = \frac{1}{N_{TDOA}} \sum_{l=1}^{N_{TDOA}} \frac{1}{\sigma} \aleph \left(\frac{\|\tau - \tau_l\|}{\sigma} \right) = \frac{1}{N_{TDOA}} \sum_{l=1}^{N_{TDOA}} \frac{1}{(2\pi\sigma^2)^{1/2}} e^{-\frac{\|\tau-\tau_l\|^2}{2\sigma^2}}, \quad (4.24)$$

where σ is the standard deviation of the Gaussian kernel. This value is computed using Silverman's rule of thumb [156],

$$\sigma^* = 0.9 N_{TDOA}^{-1/5} \cdot \min(\sigma, IQR/1.34), \quad (4.25)$$

where σ and IQR are the standard deviation and the interquartile range computed from the input data $\boldsymbol{\tau}$ respectively. In order to provide robustness to the estimation of $p(\tau)$, positive and negative extreme values are removed from $\boldsymbol{\tau}$. The remaining unknown elements in $\boldsymbol{C}$ are computed following the same procedure but replacing N_{spk} by the speaker model id number.

Density kernels are used instead of histograms to estimate the probability density because this approach does not depend on the bin width [58] and the peaks are therefore more accurately estimated.

The other unknown term in (4.18), $\boldsymbol{\beta}$, is found by maximizing the likelihood of the model parameters given the TDOAs. This maximization problem is solved using Expectation-Conditional Maximization [157].

4.3.2.1.2 Linear constraints on the standard deviation

Linear constraints on the standard deviation are fixed a priori by the vector $\mathcal{G}$. This vector is defined such that the deviation of the noise model is wider than the deviations of the speakers' model since there might be outliers with extreme TDOA values. Additionally, head movements of both speakers are assumed to be similar and also the main reason for the variance of the computed TDOAs, therefore the standard deviation of the speakers' models is the same. Hence, the linear constraints on the standard deviation are

$$\boldsymbol{\iota} = \mathcal{G}\boldsymbol{\Upsilon}, \quad (4.26)$$

$$\begin{bmatrix} 1/\sigma_B \\ 1/\sigma_1 \\ 1/\sigma_2 \\ . \\ . \\ . \\ 1/\sigma_{N_{spk}} \end{bmatrix} = \begin{bmatrix} \iota_B \\ \iota_1 \\ \iota_2 \\ . \\ . \\ . \\ \iota_{N_{spk}} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ . & . \\ . & . \\ . & . \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Upsilon_1 \\ \Upsilon_2 \end{bmatrix}, \tag{4.27}$$

$$\begin{aligned} \sigma_B &= 1/\Upsilon_1, \\ \left\{\sigma_1, \sigma_2, \cdots, \sigma_{N_{spk}}\right\} &= 1/(\Upsilon_1 + \Upsilon_2), \end{aligned} \tag{4.28}$$

since all elements of $\boldsymbol{\Upsilon}$ are non-negative, $\sigma_B \geq \left\{\sigma_1, \sigma_2, \cdots, \sigma_{N_{spk}}\right\}$.

In this case, the term $\boldsymbol{\Upsilon}$ is estimated by maximizing the likelihood of the parameters given the input data. This maximization problem is solved employing the Minorization-Maximization algorithm [157].

Additionally, variance upper and lower bounds are applied to avoid unlikely values. These variances are set to 1.25 ms and 0.03125 ms respectively, which are found experimentally.

4.3.2.2 Local modelling

In order to deal with the same speaker talking in different positions it is necessary to find the parameters of $\boldsymbol{\theta}$ for small time analysis windows of length $N_w \ll N_{TDOA}$ where the speaker is static. Assuming each speaker does not move from their position in this time analysis window, the modelling (4.17) becomes

$$\arg\max_{\boldsymbol{\theta}^v} \log \mathcal{L}(\boldsymbol{\theta}^v | \boldsymbol{\tau}^v), \tag{4.29}$$

where:

$$v = \left\{1, 2, \cdots, \left\lceil \frac{N_{TDOA}-(N_w-N_o)}{N_o} \right\rceil \right\},$$
$$\boldsymbol{\tau}^v = \{\tau_{(v-1)\cdot(N_w)+1}, \tau_{(v-1)\cdot(N_w)+2}, \cdots, \tau_{(v-1)\cdot(N_w)+N_w}\},$$

and where N_o is the number of overlapped frames and v represents the analysis window index.

The posteriors of the overlapped TDOAs is now recomputed as the average of the overlapped posteriors between both analysis windows.

4.3.2.3 Alignment

Two different alignments are needed to ensure that speaker indexes represent the same talker between two consecutive analysis windows (alignment within channel) and between the channels in the same window (alignment between channels).

4.3.2.3.1 Alignment within channel

Alignment within channel aims at finding for a given channel the correspondence of the speakers between two consecutive analysis windows. The adopted solution is based on overlapping consecutive windows as it is shown in Fig. 4.11.

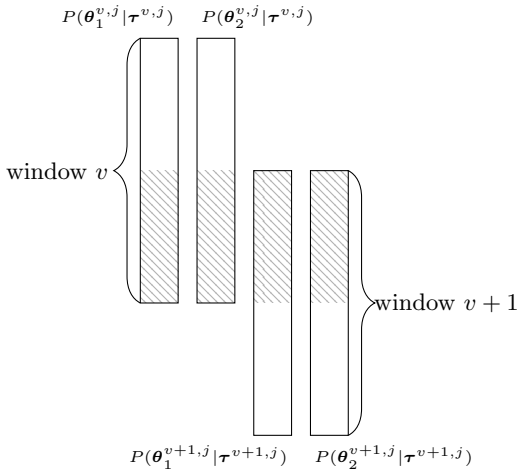

Figure 4.11: Representation of alignment within channel for the pair of microphones j and $N_{spk} = 2$.

For simplicity, the TDOAs that are common in two consecutive frames, i.e. overlapped TDOAs, are denoted as $\boldsymbol{\tau}_o$ while N_o represents the number of overlapped frames.

The alignment within channel seeks the decision vector $\boldsymbol{d}$ such that

$$\arg\max_{\boldsymbol{d}} \sum_{o=1}^{N_o} f(d_1(o), d(o)), \tag{4.30}$$

$$f(x,y) = \begin{cases} 1 & \text{if} \quad x = y \\ 0 & \text{if} \quad x \neq y \end{cases}, \tag{4.31}$$

where, assuming $N_{spk} = 2$, the vector $\boldsymbol{d}$ is defined as set of candidate vectors $\boldsymbol{d}_2$ and $\widetilde{\boldsymbol{d}_2}$ where the latter vector is the permutation of the former

$$\widetilde{\boldsymbol{d}_2} = \boldsymbol{d}_2 \pmod 2 + 1, \tag{4.32}$$

while the individual decision vectors $\boldsymbol{d}_1 = [d_1(1), d_1(2), \cdots, d_1(N_o)]$ and $\boldsymbol{d}_2 = [d_2(1), d_2(2), \cdots, d_2(N_o)]$ represent the estimated speaker indexes in the overlapped frames between the v and $v+1$ windows respectively for channel j

$$\boldsymbol{d}_1 = \arg\max_{\boldsymbol{i}} P(\boldsymbol{\theta}_i^{v,j} | \boldsymbol{\tau}_o), \tag{4.33}$$

$$\boldsymbol{d}_2 = \arg\max_{\boldsymbol{i}} P(\boldsymbol{\theta}_i^{v+1,j} | \boldsymbol{\tau}_o). \tag{4.34}$$

If $\boldsymbol{d} = \widetilde{\boldsymbol{d}_2}$, then $P(\boldsymbol{\theta}_1^{v+1,j} | \boldsymbol{\tau}^{v+1,j})$ and $P(\boldsymbol{\theta}_2^{v+1,j} | \boldsymbol{\tau}^{v+1,j})$ are swapped. This is applicable for N_{spk}= 2, although it can be extended to any value of N_{spk} by creating $\boldsymbol{d}$ such that it contains $N_{spk}!$ vectors with all possible decision permutations. In this case, same decisions within each vector permute to the same values.

4.3.2.3.2 Alignment between channels

The alignment between channels verifies whether $\boldsymbol{\theta}_1^{v,1}$ represents the TDOAs of the speaker that is modelled with $\boldsymbol{\theta}_1^{v,j}$ or the speaker that is modelled with $\boldsymbol{\theta}_2^{v,j}$ for $j = \{2, \cdots, J\}$. This verification is carried out by finding $\boldsymbol{d}$ such that

$$\arg\max_{\boldsymbol{d}} \sum_{n=1}^{M} s(d_1(n), d(n)), \tag{4.35}$$

$$s(x,y) = \begin{cases} 1 & \text{if} \quad x = y \\ 0 & \text{if} \quad x \neq y \end{cases}, \tag{4.36}$$

where, assuming $N_{spk} = 2$, the vector $\boldsymbol{d}$ is defined as set of candidate vectors $\boldsymbol{d}_j$ and $\widetilde{\boldsymbol{d}}_j$ where the latter vector is the permutation of the former

$$\widetilde{\boldsymbol{d}}_j = \boldsymbol{d}_j \pmod 2 + 1. \tag{4.37}$$

and the individual decision vectors are

$$\boldsymbol{d}_1 = \arg\max_i P(\boldsymbol{\theta}_i^{v,1}|\boldsymbol{\tau}^{v,1}), \tag{4.38}$$

$$\boldsymbol{d}_j = \arg\max_{\boldsymbol{i}} P(\boldsymbol{\theta}_i^{v,j}|\boldsymbol{\tau}^{v,j}). \tag{4.39}$$

If $\boldsymbol{d} = \widetilde{\boldsymbol{d}}_j$, then $P(\boldsymbol{\theta}_1^{v,j}|\boldsymbol{\tau}^{v,j})$ and $P(\boldsymbol{\theta}_2^{v,j}|\boldsymbol{\tau}^{v,j})$ are swapped. Figure 4.12 displays the notation for $J = 3$. This approach can be applied to any $N_{spk} > 2$ by forming $\boldsymbol{d}$ such that it comprises $N_{spk}!$ vectors with all possible decision permutations. Again in this case, same decisions within each vector permute to the same values.

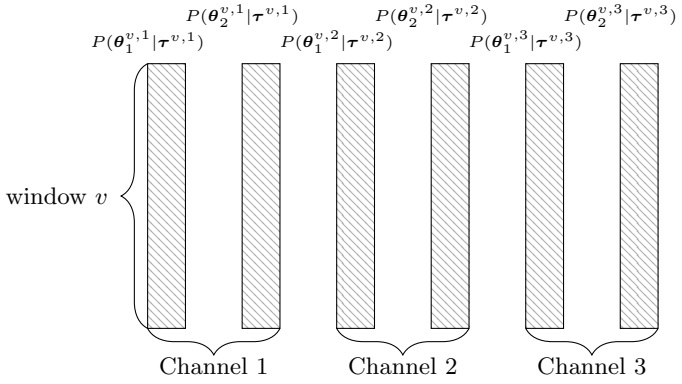

Figure 4.12: Representation of alignment between channels for window v and N_{spk}.

Both previous alignments have a complexity of $O(N_{spk}!)$, consequently the execution time rapidly increases when there are more than 3 speakers. In order to reduce this complexity, a stochastic search is performed using a GA [158] when $N_{spk} \geqslant 7$. In this case, the chromosomes encode the speaker permutations and the fitness function is

derived from (4.30) and the crossover and mutation probabilities are set empirically to 0.9 and 0.05 respectively.

4.3.2.4 Channel selection

In the local modelling process (4.29), J different models are fitted to the data which is extracted from J channels, i.e. J different pairs of microphones, and then only the optimal model is used to diarize (4.13). A priori, the pair that is closer to the speaker is likely to be the best pair but the position of speakers and microphones is unknown and additionally spurious noise can degrade the TDOAs computed in those pairs of microphones that are close to the noise source.

The channel selection aims at choosing the best pair to diarize, i.e the model that provides the lowest DER, however the labels are unknown and therefore the DER can not be directly minimized. Instead, the commonly applied metric in model selection [159] of the Bayesian Information Criterion (4.41) is used to find the optimal pair of microphones j as follows

$$\{\boldsymbol{\theta}^v, \boldsymbol{\tau}^v\} = \arg\max_j \text{BIC}(\boldsymbol{\theta}^{v,j}, \boldsymbol{\tau}^{v,j}), \tag{4.40}$$

$$\text{BIC}(\boldsymbol{\theta}, \boldsymbol{\tau}) = -2 \log \mathcal{L}(\boldsymbol{\theta}|\boldsymbol{\tau}) + N_{fp} \cdot \log(N_{TDOA}), \tag{4.41}$$

where

$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{\tau})$: likelihood of the model $\boldsymbol{\theta}$ given the data $\boldsymbol{\tau}$, i.e. $P(\boldsymbol{\tau}|\boldsymbol{\theta})$,

N_{fp} : the number of free parameters to be estimated,

N_{TDOA} : total number of TDOA samples.

The expression (4.40) selects the model that maximizes its likelihood given the TDOA estimates since the models compared in (4.40) share the same N_{fp} and N_{TDOA}.

4.3.2.5 Combination of channels

Alternatively, in this section, rather than selecting only one channel to perform Maximum A Posteriori (MAP) speaker labelling decisions, two approaches are described to combine

the information between all the available channels.

4.3.2.5.1 Maximum (MAX)

In this case MAP (4.13) is performed over all J channels

$$\arg\max_i \left(\max_j \; P(\boldsymbol{\theta}_i^j | \tau_l^j) \right), \tag{4.42}$$

where $i = \{1, \cdots, N_{spk}\}$ and $j = \{1, 2, \cdots, J\}$.

4.3.2.5.2 Average (AVG)

In this case MAP (4.13) is performed over the average of all J channels

$$\arg\max_i \sum_{j=1}^{J} \frac{1}{J} \; P(\boldsymbol{\theta}_i^j | \tau_l^j), \tag{4.43}$$

where $i = \{1, \cdots, N_{spk}\}$.

4.3.2.6 HMM

An HMM, as shown in Fig. 4.13, is implemented in order to include prior models for utterance duration and thereby potentially avoid very unlikely short utterances from one speaker [160].

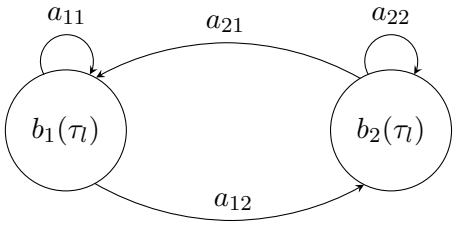

Figure 4.13: HMM architecture used for $N_{spk} = 2$.

The lowest error in the development set is achieved for $\mathcal{W}_{\text{DRR}} = 0.18$. This weight reduces the speaker error from 1.95% to 1.2%, achieving a relative improvement of 38%.

Each state of the HMM represents one speaker and the transition probabilities a_{qr} and observation probabilities b_q are computed as

$$\begin{aligned} a_{12} &= a_{21}, \\ a_{11} &= 1 - a_{12}, \\ a_{22} &= 1 - a_{21}, \\ b_1(\tau_l) &= P(\boldsymbol{\theta}_1^v|\tau_l), \\ b_2(\tau_l) &= P(\boldsymbol{\theta}_2^v|\tau_l), \end{aligned} \tag{4.44}$$

where a_{21} is computed as the ratio of TDOA frame increment over the average speaker duration. Assuming an approximate average speaker duration of 2.5 s [7] and the TDOA frame increment of 62.5 ms, then $a_{21} = 0.025$. This ratio is derived from the fact that the number of steps in the same state is geometrically distributed [161] and its expected value is $1/(1 - a_{qq})$. Therefore $1/(1 - a_{qq})$ is set to be the average speaker duration in frames. For $N_{spk} > 2$, all the states are still interconnected and the $1/(1 - a_{qq})$ is still computed as the average speaker duration in frames, however $a_{qr} = (1 - a_{qq})/(N_{spk} - 1)$.

Thus, the speaker estimate label at frame l can be extracted by applying the Viterbi algorithm

$$\arg \max_i \delta_i(l), \tag{4.45}$$

where

$$\delta_r(l) = \max_q \delta_q(l-1)\ a_{qr}\ b_r(\tau_l),$$

$$\delta_q(1) = \pi_q b_q(\tau_1),$$

and where $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_{N_{spk}}\}$ are the initial state probabilities

4.3.2.7 Confidence measure

A confidence measure CM_l that indicates the reliability of the estimated speaker index i at the time instant l can be computed directly from the problem formulation (4.13) as

$$\mathrm{CM}_l = \max_i \; P\left(\boldsymbol{\theta}_i|\tau_l\right), \tag{4.46}$$

where $i = \{1, \cdots, N_{spk}\}$ and $P\left(\boldsymbol{\theta}_i|\tau_l\right)$ are computed depending on the strategy followed to select or combine the channels.

Additionally, expressions (4.40) or (4.42) can be directly used to select the microphones with the a priori best speech signal by finding the j that maximizes these expressions.

A total of 6 different versions of the proposed method are evaluated for the speaker diarization task. The first three are described in Section 4.3.2.4, Section 4.3.2.5.1 and Section 4.3.2.5.2 and they are referred in the result section as "channel selection","MAX" and "AVG" respectively. Furthermore, another three approaches based on combining the previous versions with an HMM are assessed and termed as "channel selection+HMM","MAX+HMM" and "AVG+HMM". These combinations are accomplished by setting the HMM observation probabilities to the posteriors of each method.

4.3.3 Experimental setup

Two different databases are considered to evaluate the method presented in this work: an artificial database comprising simulated meeting scenarios and a real database with recordings from real meetings.

4.3.3.1 Simulated room impulse responses

This database is designed to test the performance of the presented method under different controlled environments, i.e. microphones/speakers positions or reverberation level. The main characteristics of this database are:

- 2 different female speakers.
- Total recording length of 28 s.
- 10 different utterances are included, 5 from each speaker.

- No additional noise.
- In order to create a recording that represents conversational speech 3 utterances out of the 10 utterances are relatively short: 0.26 s , 0.17 s, and 0.45 s.
- Figure 4.14 shows the used setup. The dimensions of the room and the position of speakers and microphones are displayed in Table 4.5. There are 2 positions for each speaker and the microphone positions and room size are fixed. All RIRs are generated using the randomized image method [72].

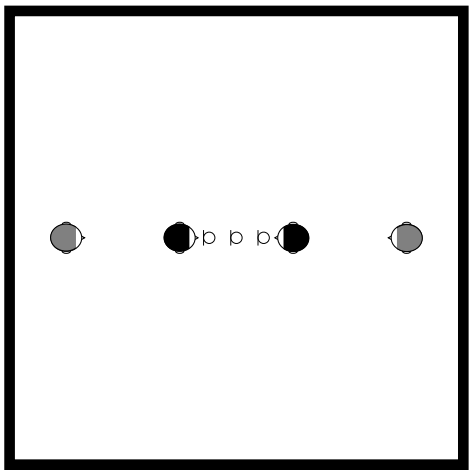

Figure 4.14: Sketch of the simulated room indicating the positions of the microphones and speakers. Microphones are fixed whereas speakers are located in two different places which are represented with black hair and gray hair heads.

- The values of three room acoustic parameters, T_{60}, C_{50} and DRR, for each setup are displayed in Table 4.6.

Setup id	Room size	Speaker 1 position	Speaker 2 position	Mic. 1 position	Mic. 2 position	Mic. 3 position
1	[4,4,2.5]	[1.5,2,1]	[2.5,2,1]	[2.25,2,0.5]	[2,2,0.5]	[1.75,2,0.5]
2	[4,4,2.5]	[0.5,2,1]	[2.5,2,1]	[2.25,2,0.5]	[2,2,0.5]	[1.75,2,0.5]
3	[4,4,2.5]	[1.5,2,1]	[3.5,2,1]	[2.25,2,0.5]	[2,2,0.5]	[1.75,2,0.5]
4	[4,4,2.5]	[0.5,2,1]	[3.5,2,1]	[2.25,2,0.5]	[2,2,0.5]	[1.75,2,0.5]

Table 4.5: Description of the setup configurations according to the positions of the speakers and microphones displayed in Fig. 4.14. The values within the squared brackets represent x, y and z axis values.

Label	$\mathbf{T_{60}}$ (s)	$\mathbf{C_{50}}$ (dB)	DRR (dB)	Setup id
R_1P_1	0	∞	∞	1
R_1P_2	0	∞	∞	2
R_1P_3	0	∞	∞	3
R_1P_4	0	∞	∞	4
R_2P_1	0.2	[21.29, 23.58]	[-2.40, 3.59]	1
R_2P_2	0.2	[18.96, 24.11]	[-7.02, 0.81]	2
R_2P_3	0.2	[18.92, 22.94]	[-7.04, -0.13]	3
R_2P_4	0.2	[18.55, 20.45]	[-8.72, -2.28]	4
R_3P_1	0.4	[8.47,11.02]	[-6.71, -0.54]	1
R_3P_2	0.4	[6.98, 10.51]	[-9.38, -0.36]	2
R_3P_3	0.4	[7.62, 12.44]	[-9.32, -0.57]	3
R_3P_4	0.4	[7.47, 8.80,]	[-11.98, -7.26]	4
R_4P_1	0.6	[4.89, 6.48]	[-9.01, -4.00]	1
R_4P_2	0.6	[3.74, 6.24]	[-12.29, -2.67]	2
R_4P_3	0.6	[3.67, 6.54]	[-13.21, -3.39]	3
R_4P_4	0.6	[3.48, 4.58]	[-14.32, -8.39]	4
R_5P_1	0.8	[2.24, 4.01]	[-11.20, -3.76]	1
R_5P_2	0.8	[1.62, 4.07]	[-13.03, -6.71]	2
R_5P_3	0.8	[1.65, 4.34]	[-12.40, -2.96]	3
R_5P_4	0.8	[1.57, 1.97]	[-13.72, -9.93]	4
R_6P_1	1	[0.93, 3.26]	[-12.37, -3.09]	1
R_6P_2	1	[-0.13, 2.62]	[-15.13, -7.37]	2
R_6P_3	1	[-0.18, 2.61]	[-15.79, -6.32]	3
R_6P_4	1	[-0.02, 0.63]	[-16.44, -12.64]	4

Table 4.6: Label assigned to each evaluation condition. The setup id is shown in Table 4.5. The quantities within the squared brackets represent the maximum and minimum values obtained with the three different microphones and two speakers.

4.3.3.2 Real meeting corpus

This database comprises the conference room meetings from NIST RT-05 Evaluation Corpora [153]. This corpora is part of the National Institute of Standards and Technology transcription series launched to promote and improve speech research tools. The meetings included in this corpora provide real scenarios with highly interactive discussions between multiple speakers. The proposed algorithm is evaluated on the multi distant microphones, usually placed on a table between participants, instead of the individual head microphones. This configuration is chosen due to being more flexible since the microphones can be placed anywhere in the room, which makes it also more challenging from the signal processing

point of view [153].

Table 4.7 outlines the details of the recordings used in this evaluation. There are in total 10 recordings from different sites: AMI (Augmented Multi-party Interaction project), CMU (Carnegie Mellon University Interactive Systems Laboratory), ICSI, NIST, and VT (Virginia Tech). The length of this evaluation set is approximately 2 hours, with 12 minutes for each recording.

Label	**File name**	**Number of speakers**	**Number of microphones**	**Duration** (minutes)
AMI1	AMI_20041210-1052	4	8	12.2
AMI2	AMI_20050204-1206	4	8	11.9
CMU1	CMU_20050228-1615	4	3	12.0
CMU2	CMU_20050301-1415	4	3	12.0
ICSI1	ICSI_20010531-1030	7	6	12.2
ICSI2	ICSI_20011113-1100	9	6	12.0
NIST1	NIST_20050412-1303	10	7	12.1
NIST2	NIST_20050427-0939	4	7	11.9
VT1	VT_20050304-1300	5	2	12.0
VT2	VT_20050318-1430	5	2	12.1

Table 4.7: Summary of RT05 evaluation set.

4.3.3.3 Evaluation

The presented approaches are analysed and compared to the baselines introduced in Section 4.3.1 in terms of speaker time error as described in Section 4.2.3.2.

4.3.4 Results

4.3.4.1 Simulated room impulse responses

In this case a time analysis window of 15 s with 50% overlap is used for the local modelling. The analysis window size was chosen empirically as a trade-off between the minimum amount of data to accurately perform the modelling and the maximum period of time where the speakers' position are fixed. Although in this database the speakers' position are fixed, the local modelling is applied in order to analyse its performance.

Figure 4.15 shows the performance achieved for each simulated evaluation scenario. The speaker labels for channel selection approach, MAX and AVG are obtained following Section 4.3.2.4, expression (4.42) and expression (4.43) respectively. Moreover, channel selection + HMM, MAX+HMM and AVG+HMM are achieved following expression (4.42), where the observation probabilities correspond to the posteriors used to estimate the speakers. Figure 4.15 suggests that the methods are dependent of the level of reverberation in the room. This dependence is mainly due to the sound reflections occurring in reverberant environment which cause inaccurate TDOA estimates.

Figure 4.16 shows the waveform of the simulated recording for R_1P_1 and the diarization result using MAX. In this case the diarization is performed without any errors, assuming perfect VAD.

Figure 4.17 shows the speaker error achieved on average with the different presented approaches and the OG baseline. This baseline achieves a speaker error rate of 15.84% which is computed for a given recording as the average of the errors obtained on each TDOA channel. However, when applying the approaches in this section lower errors are obtained. The best performance is achieved with AVG+HMM where the speaker error rate is 6.53%. Thus, the proposed method outperforms on average the OG baseline even though it does not assume any specific position of the microphones and, unlike the OG baseline, it can be employed in scenarios with more than 2 speakers. Moreover, Fig. 4.17 suggests that applying an HMM to the different approaches does not improve the results.

4.3.4.2 Real meeting corpus

In order to optimize the results and since the speakers are assumed to be still in these meeting recordings, the time analysis window used for this evaluation set is the size of the whole recording, hence no local modelling is performed. Moreover, the evaluation results are obtained using only the maximum number of speakers, i.e. 10, due to DiarTK limitations when setting the correct number of speakers. Consequently, in the case of the system proposed in this chapter, the number of speakers is set to 10 for each of the test recordings. Likewise, in the case of DiarTK system the maximum number of speakers is

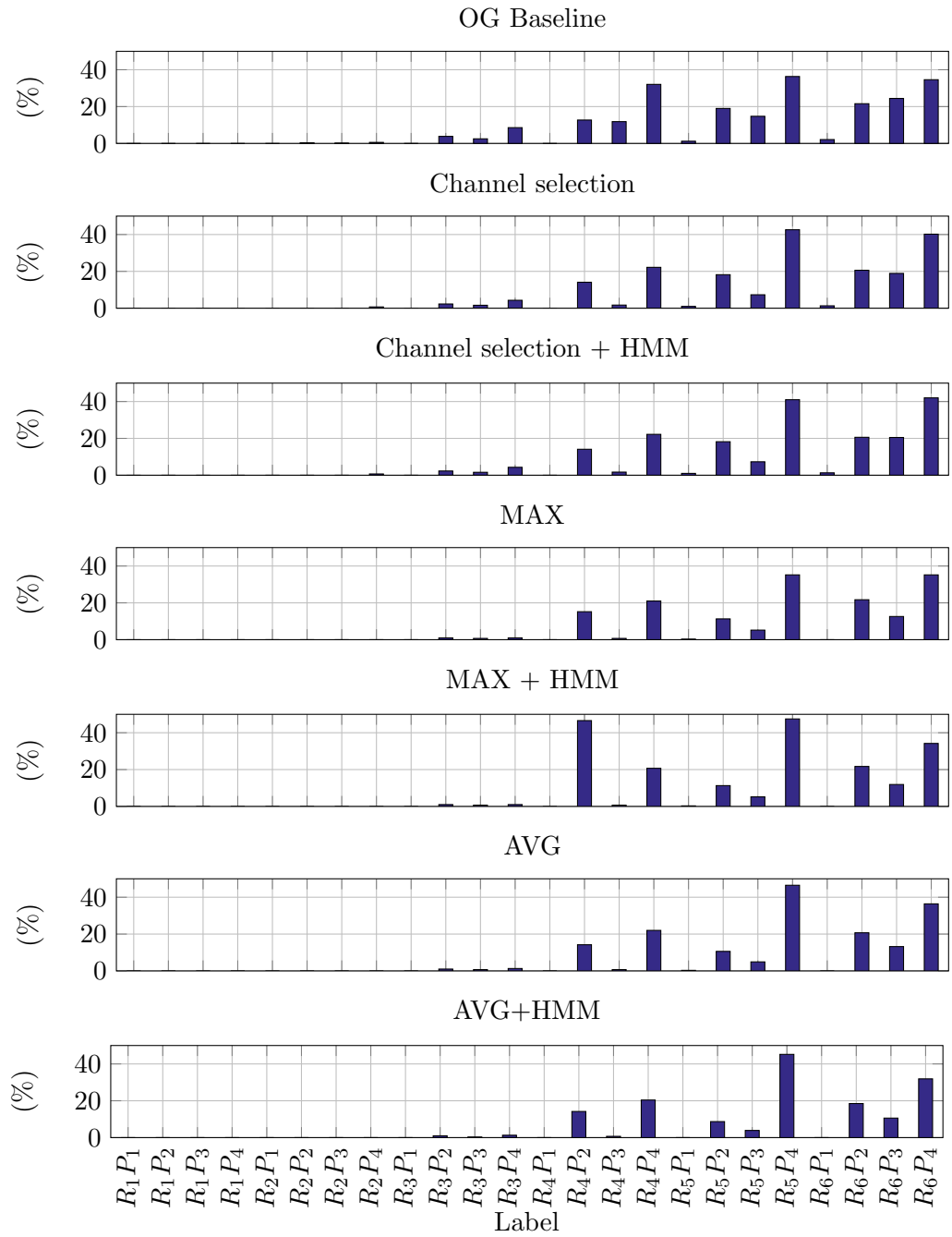

Figure 4.15: Speaker error obtained with the proposed method for each simulated evaluation subset shown in Table 4.6.

set to 10. Thus both systems can be compared in the same test conditions.

Figure 4.18 shows the speaker error rate achieved on average with the baseline, i.e. DiarTK [133], and with the different approaches proposed in this work. The figure indicates firstly that the baseline DiarTK provides on average worse performance than any

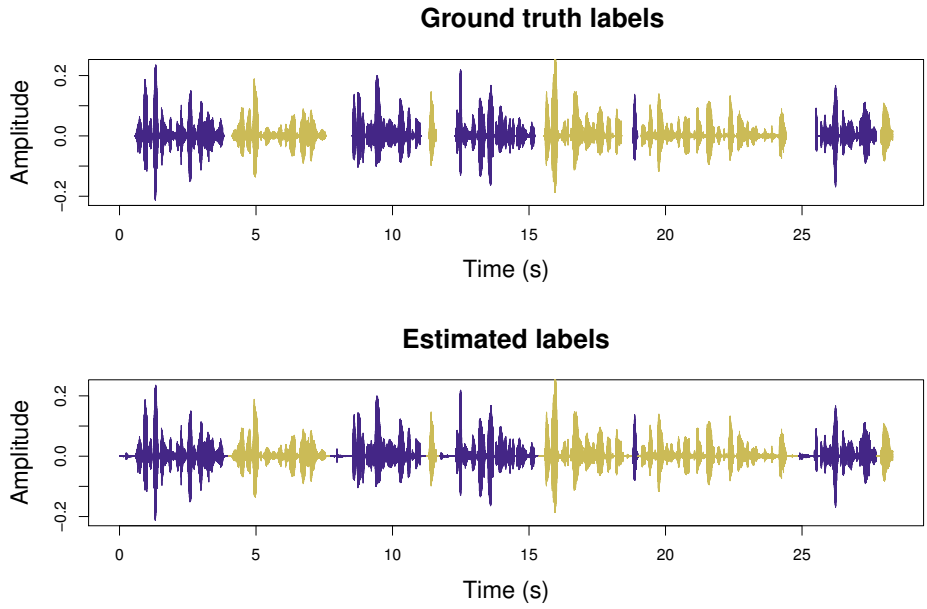

Figure 4.16: Example of diarization result. Blue and yellow segments represent different speakers. Blank spaces in the ground truth (top plot) represent silences.

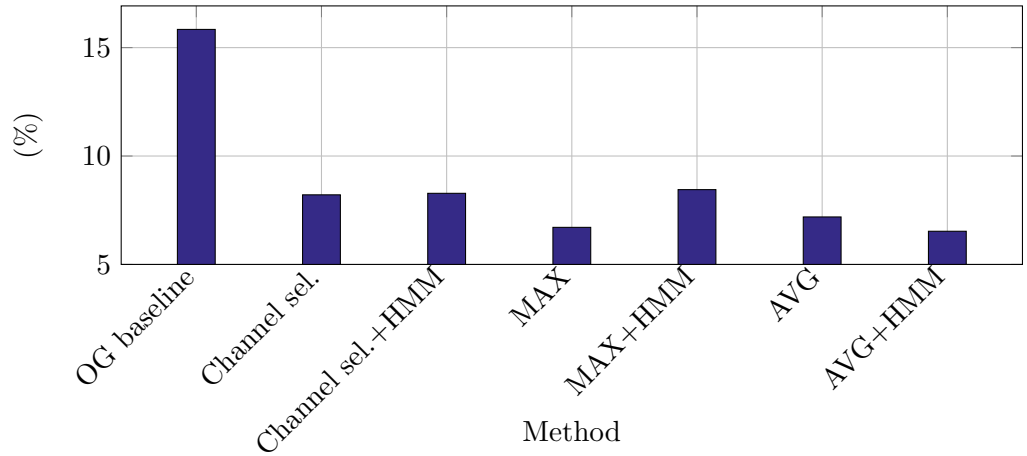

Figure 4.17: Comparison of the average speaker error achieved with the different approaches on the simulated data.

of the approaches proposed, i.e 39.65% speaker error rate. Secondly, the best approach on this evaluation set is achieved with AVG+HMM, i.e 23.71% speaker error rate. Thirdly, the incorporation of HMM into the system reduces on average the speaker error by 1.4% absolute.

It should be noted that DiarTK estimates the number of speakers internally through

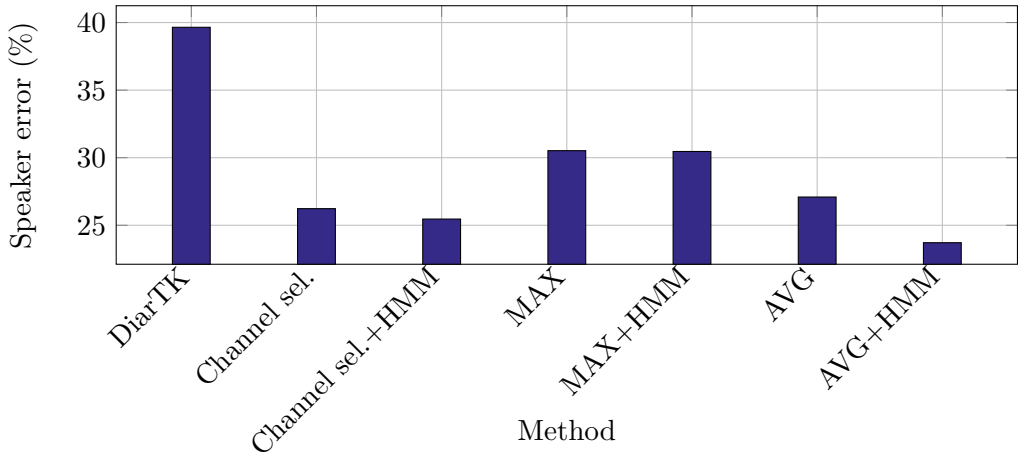

Figure 4.18: Comparison of the average speaker error achieved with the different approaches on the RT05 database.

an agglomerative process, see Section 4.2.1, thus it tries to iteratively reduce the maximum number of speakers set to 10 until the optimal is reached. In contrast, the proposed method does not attempt to reduce the number of speakers and consequently it builds 10 Gaussian models for each TDOA stream available in each recording. By setting the correct number of speakers in the latter method, the speaker error yields to a further decrease to 17.06%. This outcome suggests that the proposed method is not very sensitive to overestimating the number of speakers.

Figure 4.19 shows the breakdown of the average speaker error for each method. The best approach outperforms the baseline in each recording expect in the recording NIST2. The error rates achieved with recordings VT1 and VT2 are consistently the same for all the proposed approaches. This is due to the fact that these two recordings comprise two microphone channels, thereupon it is only possible to compute one TDOA stream and it is not feasible to perform any combination of channels. This issue is also reflected in the higher errors compared to the remainder recordings. In general the performance of the different methods is independent of the number of speakers. The error of NIST1 which comprises 10 speakers is relatively high while the error of ICSI2 which comprises 9 speaker is relatively low. It should be noted that NIST1 contains one speaker talking over a conference phone which may be affecting the results and reducing the performance of the diarization methods.

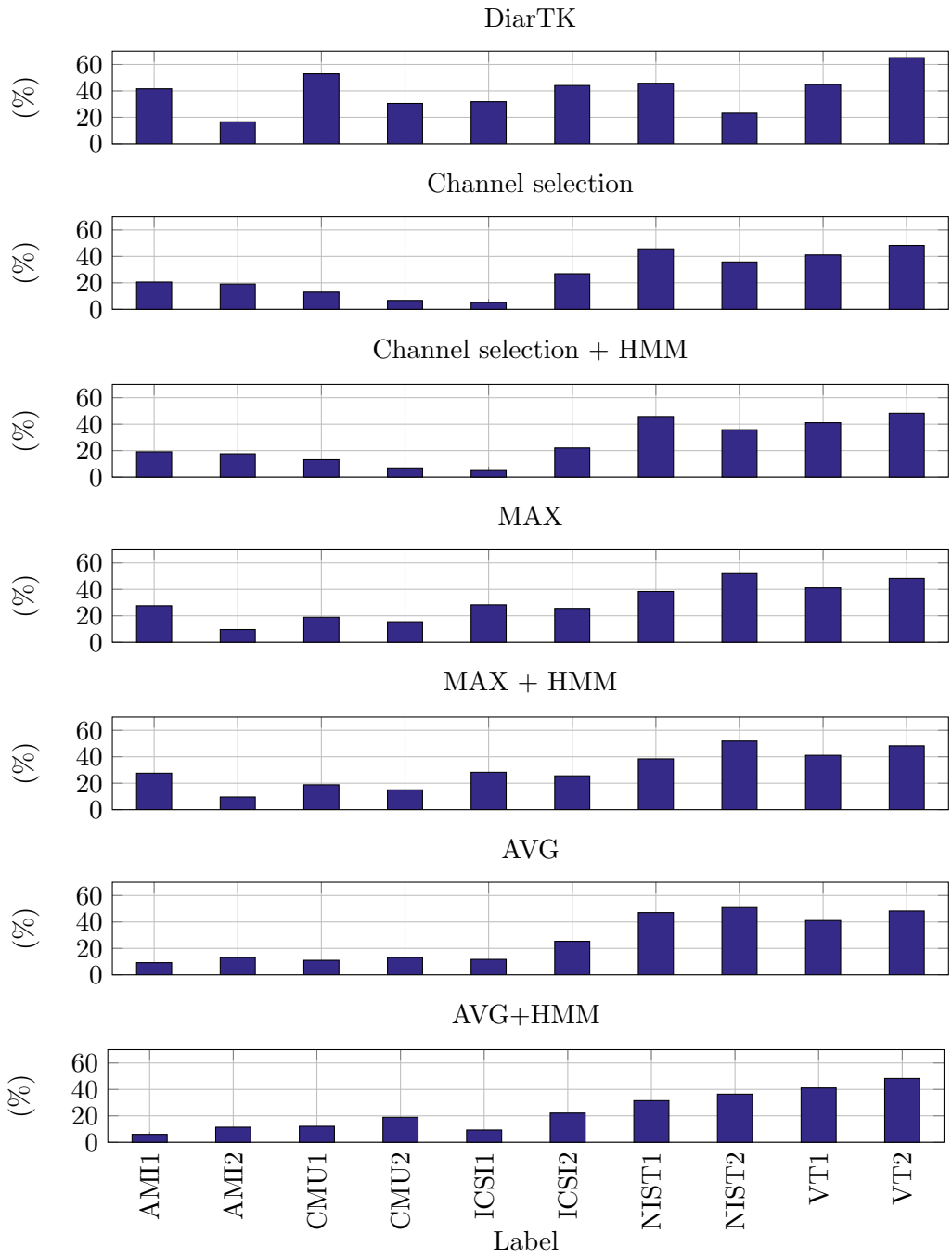

Figure 4.19: Speaker error achieved with the proposed method for each RT05 evaluation subset shown in Table 4.7.

Figure 4.20 shows the confidence measures obtained with AVG+HMM, assuming the exact number of speakers is known, against the speaker label estimation accuracy. This accuracy is computed for each confidence measure band and for each of the 10 recordings in RT05. The width of the confidence measure band is set to 0.1. Additionally, the

average of these points is computed and represented with a black line. It shows an upward trend in the confidence measure as the accuracy increases. On average the accuracy of the estimation with a confidence lower than 0.5 is 55% while the estimations with a confidence higher than 0.5 provide 77% accuracy.

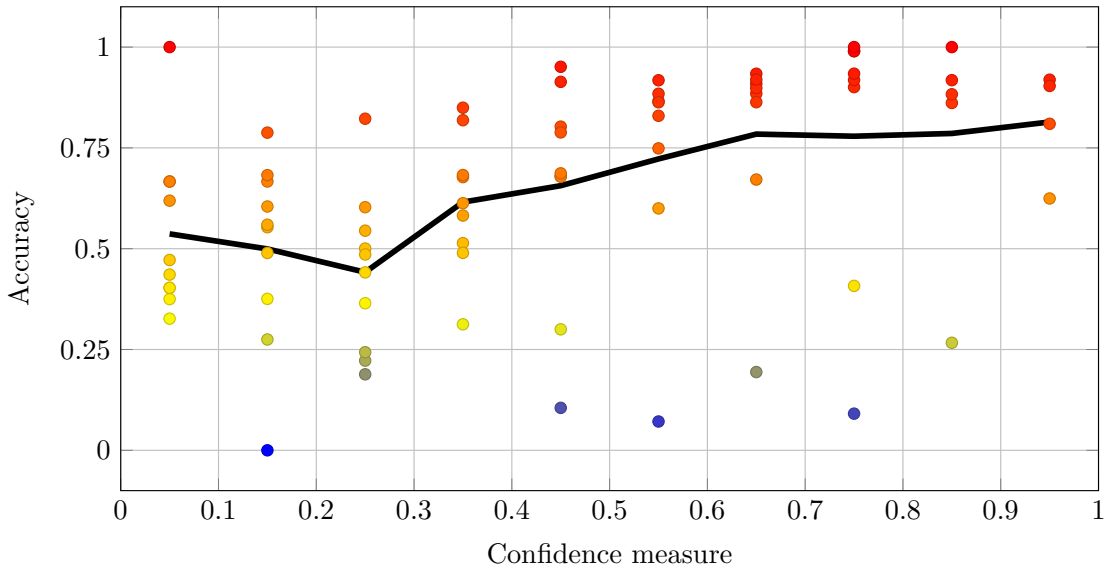

Figure 4.20: Accuracy of speaker label estimations grouped according to the confidence measure range. Each point represents the accuracy achieved in each RT05 recording. The black line represents the average of these points for each confidence measure range.

4.3.5 Conclusions

A method to perform diarization based on modelling only spatial features, i.e. TDOA estimates, in an unsupervised manner with specific constraints has been investigated.

This method has been proven to outperform a state-of-the-art method, i.e. DiarTK, under the same test conditions. On average, the method reduces the speaker error rate by approximately 9.3% (58.8% relative error reduction) on simulated data with respect to the OG baseline. Furthermore, the method has been evaluated on real meeting recordings from RT05 where it reduces the speaker error rate of the baseline DiarTK by 15.9% (40.2% relative error reduction). Further error reduction of the proposed method has been achieved by employing the exact number of speakers in each recording. In this case,

the speaker error rate is reduced to 17%. The number of speakers in recordings could potentially be estimated with external algorithms [162].

Additionally, confidence measures of the speaker estimations have been explored showing that the a posteriori probability extracted from the proposed method is related to the accuracy of the estimations.

Chapter 5

Conclusion

In this chapter the summary of this thesis is first presented in Section 5.1 and then some further research suggestions are outlined in Section 5.2.

5.1 Summary

The aim of this thesis was to robustly perform speech recognition and diarization in reverberant environments. Throughout the chapters different challenges of the thesis were tackled.

In Chapter 2 the **measure of reverberation most correlated with ASR was found** and **a method to non-intrusively estimate measures of reverberation from single-channel recordings was developed**. The full frequency-band C_{50} was shown to be the most relevant measure of reverberation to predict phoneme recognition in terms of correlation and mutual information. As a result, a non-intrusive method (NIRA) to estimate C_{50} from reverberant speech was proposed and evaluated over a database of $\approx$93 hours of reverberant speech. In addition, prediction intervals and confidence measures of the C_{50} estimates were investigated to provide an indication of the accuracy of the estimates. Finally, the NIRA framework was adapted to estimate full-band DRR and T_{60} from single-channel reverberant speech. This configuration was evaluated within the ACE Challenge achieving the best performance for single-channel DRR estimation and second

best performance over all 27 DRR estimation methods submitted to the challenge [92]. However, NIRA framework was shown to less accurate estimating T_{60} due to generalization issues.

In Chapter 3 a method that **leverages reverberation measure estimates to perform reverberant speech recognition** was designed. The degradation in phoneme recognition was analyzed with different ASR under several reverberant conditions and the confusability factor was proposed to characterize the confusion of recognizing the phonemes depending on the reverberation level. Then, this confusability factor was employed to improve the speech recognition performance in reverberant environments, however the improvement achieved in WER was subtle. Finally, a reverberant speech recognition method was proposed based on including C_{50} as an additional feature in the ASR followed by the dimension reduction technique HLDA to match the original feature vector dimension and to perform acoustic model selection to reduce the mismatch between data and ASR acoustic models. While this method did not reduce the WER close to 0 in all evaluation subsets, it was shown to be complementary to other techniques such as CMLLR.

In Chapter 4 **the use of reverberation measure estimates to perform single-channel diarization in reverberant environments was investigated** and **a multi-channel based diarization method robust to distant-talk was designed**. Estimations of DRR were shown to be a feature that can discriminate different speakers located in a room at different positions relative to the microphone and they were employed, along with MFCC features, to improve diarization performance compared to using only MFCC features. Additionally, a method to perform diarization on multi-channel meeting recordings was investigated employing TDOA estimates and Gaussian models created in an unsupervised manner with specific constraints. This method outperformed a state-of-the-art method, i.e. DiarTK, under the same test conditions and it was shown to provide low diarization performance in highly reverberant environments, i.e $T_{60} \geq 0.8$ s. Furthermore, a confidence measure for the speaker label estimation was analysed and tested.

5.2 Future work

In this section a few potential lines of research related with this thesis are proposed. They either further develop proposed methods or validate these methods in new contexts.

- **Evaluation of dereverberation algorithms with NIRA framework**. On the one hand, Chapter 2 showed that NIRA can successfully be used to estimate measures of reverberation such as C_{50} or DRR. On the other hand, dereverberation algorithms aim to reduce the level of reverberation in the signal and increasing thus C_{50} or DRR. Therefore, NIRA could potentially be used to non-intrusively evaluate the reverberation reduction achieved in the processed signal.

- **Evaluation of the model switching method on a DNN-based ASR system**: In Section 3.4 a model switching method was proposed to perform reverberant speech recognition. However, DNN-based ASR can capture various characteristics of reverberant speech in different reverberant environments, therefore future work could address the usefulness of the proposed method in such systems.

- **Incorporation of speech features in the multi-channel diarization system proposed in Section 4.3**. The proposed multi-channel diarization method is based on modelling the TDOA features in a robust manner by setting mean and variance constraints in the EM method. These features exploit spatial characteristics of the audio recording, however no discriminative speech information, such as MFCC, is employed in the method. Therefore, the method could benefit from using this information type along with TDOAs. One potential approach to accomplish this task is to model MFCC features using unconstrained EM independently of the proposed TDOA modelling. Then, both models can be merged by adding another dimension to the MFCC model with the TDOA model estimated parameters. Finally, the decoding can be performed in the same manner where $\boldsymbol{\theta}$ represents the new merged model.

Bibliography

[1] H. Kuttruff, *Room Acoustics*, Taylor & Francis, London, fifth edition, 2009.

[2] T. H. Falk and W.-Y. Chan, "Temporal dynamics for blind measurement of room acoustical parameters," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 4, pp. 978–989, 2010.

[3] J. Wen, N. D. Gaubitch, E. A. P. Habets, T. Myatt, and P. A. Naylor, "Evaluation of speech dereverberation algorithms using the MARDY database," in *Proc. Intl. Workshop Acoust. Echo Noise Control (IWAENC)*, Paris, France, 2006.

[4] S. A. Gelfand and S. Silman, "Effects of small room reverberation upon the recognition of some consonant features," *The Journal of the Acoustical Society of America*, vol. 66, no. 1, pp. 22–29, 1979.

[5] T. H. Falk, C. Zheng, and W.-Y. Chan, "A non-intrusive quality and intelligibility measure of reverberant and dereverberated speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 7, pp. 1766–1774, 2010.

[6] T. Yoshioka, A. Sehr, M. Delcroix, K. Kinoshita, R. Maas, T. Nakatani, and W. Kellermann, "Making machines understand us in reverberant rooms: Robustness against reverberation for automatic speech recognition," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 114–126, 2012.

[7] X. Anguera Miro, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: a review of recent research," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 2, pp. 356–370, 2012.

[8] P. Peso Parada, D. Sharma, J. Lainez, D. Barreda, T. van Waterschoot, and P. A. Naylor, "A single-channel non-intrusive C50 estimator correlated with speech recognition performance," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 4, pp. 719–732, April 2016.

[9] P. Peso Parada, D. Sharma, P. A. Naylor, and T. van Waterschoot, "Reverberant speech recognition exploiting clarity index estimation," *EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, 2015.

[10] P. Peso Parada, D. Sharma, T. van Waterschoot, and P. A. Naylor, "Confidence measures for non-intrusive estimation of speech clarity index," *The Journal of the Audio Engineering Society*, 2016, Submitted.

[11] A. H. Moore, P. Peso Parada, and P. A. Naylor, "Speech enhancement evaluation using speech recognition," *Computer Speech and Language*, 2016, Submitted.

[12] P. Peso Parada, D. Sharma, and P. A. Naylor, "Non-intrusive estimation of the level of reverberation in speech," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 4718–4722.

[13] P. Peso Parada, D. Sharma, P. A. Naylor, and T. van Waterschoot, "Single-channel reverberant speech recognition using C50 estimation," in *Proc. REVERB Challenge*, Florence, Italy, May 2014.

[14] P. Peso Parada, D. Sharma, J. Lainez, D. Barreda, P. A. Naylor, and T. van Waterschoot, "A quantitative comparison of blind C50 estimators," in *Proc. Intl. Workshop Acoust. Signal Enhancement (IWAENC)*, Juan les Pins, France, September 2014, pp. 298–302.

[15] P. Peso Parada, D. Sharma, P. A. Naylor, and T. van Waterschoot, "Reverberant speech recognition: A phoneme analysis," in *Signal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on*. IEEE, December 2014, pp. 567–571.

[16] M. Hu, P. Peso Parada, D. Sharma, S. Doclo, T. van Waterschoot, M. Brookes, and P. A. Naylor, "Single-channel speaker diarization based on spatial features," in

Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. IEEE, October 2015, pp. 1–5.

[17] P. Peso Parada, D. Sharma, T. van Waterschoot, and P. A. Naylor, "Evaluating the non-intrusive room acoustics algorithm with the ACE challenge," in *ACE Challenge Workshop, a satellite event of IEEE-WASPAA 2015*, October 2015.

[18] P. Peso Parada, D. Sharma, P. A. Naylor, and T. van Waterschoot, "Analysis of prediction intervals for non-intrusive estimation of speech clarity index," in *Audio Engineering Society Conference: 60th International Conference: DREAMS (Dereverberation and Reverberation of Audio, Music, and Speech)*, February 2016.

[19] D. Sharma, P. A. Naylor, and P. Peso Parada, "Method for non-intrusive acoustic parameter estimation," Patent. U.S. 20150073780. Mar. 2015.

[20] P. Peso, D. Sharma, P. A. Naylor, and U. Jost, "Microphone selection and multi-talker segmentation with application to ambient automatic speech recognition (ASR)," U.S. Provisional Pat. Ser. No. 62/394,286, filled. September. 2016.

[21] J. Eaton, N. D. Gaubitch, A. H. Moore, and P. A. Naylor, "The ACE Challenge - corpus description and performance evaluation," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, October 2015, pp. 1–5.

[22] P. A. Naylor and N. D. Gaubitch, Eds., *Speech Dereverberation*, Springer, London, 2010.

[23] M. Karjalainen, P. Ansalo, A. Mäkivirta, T. Peltonen, and V. Välimäki, "Estimation of modal decay parameters from noisy response measurements," *Journal of the Audio Engineering Society*, vol. 50, no. 11, pp. 867–878, 2002.

[24] R. Stanton and M. Brookes, "Speech dereverberation in the STFT domain," Tech. Rep., Imperial College London, June 2013.

[25] J. M. F. del Vallado, A. A. de Lima, T. d. M. Prego, and S. L. Netto, "Feature analysis for the reverberation perception in speech signals," in *Proc. IEEE Interna-*

tional Conference on Acoustics, Speech and Signal Processing (ICASSP), Vancouver, Canada, 2013, pp. 8169–8173.

[26] T. Fukumori, M. Morise, and T. Nishiura, "Performance estimation of reverberant speech recognition based on reverberant criteria RSR-D_n with acoustic parameters," in *Proc. INTERSPEECH*, Makuhari, Japan, 2010, pp. 562–565.

[27] A. Tsilfidis, I. Mporas, J. Mourjopoulos, and N. Fakotakis, "Automatic speech recognition performance in different room acoustic environments with and without dereverberation preprocessing," *Computer Speech & Language*, vol. 27, no. 1, pp. 380–395, 2013.

[28] A. Brutti and M. Matassoni, "On the use of early-to-late reverberation ratio for asr in reverberant environments," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 4638–4642.

[29] A. Sehr, E. A. P. Habets, R. Maas, and W. Kellermann, "Towards a better understanding of the effect of reverberation on speech recognition performance," in *Proc. Intl. Workshop Acoust. Echo Noise Control (IWAENC)*, Tel Aviv, Israel, 2010.

[30] L. Couvreur, C. Ris, and C. Couvreur, "Model-based blind estimation of reverberation time: application to robust ASR in reverberant environments.," in *Proc. INTERSPEECH*, Aalborg, Denmark, 2001, pp. 2635–2638.

[31] J. Liu and G.-Z. Yang, "Robust speech recognition in reverberant environments by using an optimal synthetic room impulse response model," *Speech Communication*, vol. 67, pp. 65–77, 2015.

[32] A. Mohammed, M. Matassoni, H. Maganti, and M. Omologo, "Acoustic model adaptation using piece-wise energy decay curve for reverberant environments," in *Proc. of the 20th European Signal Processing Conference (EUSIPCO)*, Bucharest, Romania, 2012, pp. 365–369.

[33] R. Gomez and T. Kawahara, "Dereverberation based on wavelet packet filtering for robust automatic speech recognition," in *Proc. INTERSPEECH*, Portland, USA, 2012, pp. 1243–1246.

[34] H. Löllmann, E. Yilmaz, M. Jeub, and P. Vary, "An improved algorithm for blind reverberation time estimation," in *Proc. Intl. Workshop Acoust. Echo Noise Control (IWAENC)*, Tel Aviv, Israel, 2010, pp. 1–4.

[35] J. Eaton, N. D. Gaubitch, and P. A. Naylor, "Noise-robust reverberation time estimation using spectral decay distributions with reduced computational cost," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vancouver, Canada, 2013, pp. 161–165.

[36] P. Kendrick, T. J. Cox, F. F. Li, Y. Zhang, and J. A. Chambers, "Monaural room acoustic parameters from music and speech," *The Journal of the Acoustical Society of America*, vol. 124, no. 1, pp. 278–287, 2008.

[37] F. Xiong, S. Goetze, and B. T. Meyer, "Estimating room acoustic parameters for speech recognizer adaptation and combination in reverberant environments," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 5522–5526.

[38] B. Dumortier and E. Vincent, "Blind RT60 estimation robust across room sizes and source distances," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 5187–5191.

[39] E. Georganti, J. Mourjopoulos, and S. van de Par, "Room statistics and direct-to-reverberant ratio estimation from dual-channel signals," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 4713–4717.

[40] C. S. J. Doire, M. Brookes, P. A. Naylor, D. Betts, C. M. Hicks, M. A. Dmour, and S. H. Jensen, "Single-channel blind estimation of reverberation parameters," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[41] T. de M. Prego, A. A. de Lima, R. Zambrano-Lopez, and S. L. Netto, "Blind estimators for reverberation time and direct-to-reverberant energy ratio using subband speech decomposition," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, October 2015, pp. 1–5.

[42] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, et al., "The Kaldi speech recognition toolkit," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, Hawaii, USA, 2011, pp. 1–4.

[43] ITU-T, "Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs," February 2001.

[44] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Phys. Rev. E*, vol. 69, pp. 066138, June 2004.

[45] S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland, *The HTK Book, version 3.4*, Cambridge University Engineering Department, Cambridge, UK, 2006.

[46] Y. Wang and M. Brookes, "Speech enhancement using a modulation domain Kalman filter post-processor with a Gaussian mixture noise model," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 7024–7028.

[47] J. Andén and S. Mallat, "Deep scattering spectrum," *IEEE Transactions on Signal Processing*, vol. 62, no. 16, pp. 4114–4128, August 2014.

[48] H. Hermansky, "Modulation spectrum in speech processing," in *Signal Analysis and Prediction*, pp. 395–406. Springer, 1998.

[49] R. McEachern, "How the ear really works," in *Proc. IEEE-SP International Symposium Time-Frequency and Time-Scale Analysis*, Oct 1992, pp. 437–440.

[50] D. Sharma, *Speech assessment and characterization for law enforcement applications*, Ph.D. thesis, Imperial Collage London, 2012.

[51] I. V. McLoughlin, "Line spectral pairs," *Signal Processing*, vol. 88, no. 3, pp. 448–467, 2008.

[52] S. Gonzalez and M. Brookes, "A pitch estimation filter robust to high levels of noise (PEFAC)," in *Proc. of the 19th European Signal Processing Conference (EUSIPCO)*, Barcelona, Spain, 2011, pp. 451–455.

[53] D. Sharma, G. Hilkhuysen, N. D. Gaubitch, P. A. Naylor, M. Brookes, and M. Huckvale, "Data driven method for non-intrusive speech intelligibility estimation," in *Proc. of the 18th European Signal Processing Conference (EUSIPCO)*, Aalborg, Denmark, 2010, pp. 1899–1903.

[54] L. Cohen, *Time-frequency analysis*, vol. 299, Prentice hall, 1995.

[55] E. Gopi, *Algorithm collections for digital signal processing applications using matlab*, Springer Science & Business Media, 2007.

[56] ITU-T, "Objective measurement of active speech level," March 1993.

[57] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, CRC Press, Florida, 1984.

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

[59] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, John Wiley & Sons, New York, second edition, 2001.

[60] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, November 2012.

[61] A. Ng, "Sparse autoencoder," *Stanford University, CS294A Lecture notes*, 2011.

[62] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[63] T. K. Leen and G. B. Orr, "Optimal stochastic search and adaptive momentum," in *Proc. Advances in neural information processing systems (NIPS)*, Denver, USA, 1994, pp. 477–484.

[64] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vancouver, Canada, May 2013, pp. 6645–6649.

[65] F. Weninger, S. Watanabe, J. Le Roux, J. R. Hershey, Y. Tachioka, J. Geiger, B. Schuller, and G. Rigoll, "The MERL/MELCO/TUM system for the REVERB Challenge using Deep Recurrent Neural Network Feature Enhancement," in *Proc. REVERB challenge*, Florence, Italy, 2014.

[66] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. INTERSPEECH*, Makuhari, Japan, 2010, pp. 1045–1048.

[67] Z. Zhang, J. Pinto, C. Plahl, B. Schuller, and D. Willett, "Channel mapping using bidirectional long short-term memory for dereverberation in hands-free voice controlled devices," *IEEE Transactions on Consumer Electronics*, vol. 60, no. 3, pp. 525–533, August 2014.

[68] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 6, no. 02, pp. 107–116, 1998.

[69] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[70] F. Weninger, J. Bergmann, and B. Schuller, "Introducing CURRENNT–the Munich open-source CUDA RecurREnt Neural Network Toolkit," *Journal of Machine Learning Research*, vol. 15, 2014.

[71] J. S. Garofolo, "Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database," Technical report, National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, December 1988.

[72] E. De Sena, N. Antonello, M. Moonen, and T. van Waterschoot, "On the modeling of rectangular geometries in room acoustic simulations," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 4, pp. 774–786, April 2015.

[73] A. Varga and H. J. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Communication*, vol. 12, no. 3, pp. 247–251, 1993.

[74] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, E. Habets, R. Haeb-Umbach, V. Leutnant, A. Sehr, W. Kellermann, R. Maas, S. Gannot, and B. Raj, "The REVERB challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, USA, 2013, pp. 1–4.

[75] R. Stewart and M. Sandler, "Database of omnidirectional and B-format room impulse responses," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Dallas, USA, March 2010, pp. 165–168.

[76] J. K. Nielsen, J. R. Jensen, S. H. Jensen, and M. G. Christensen, "The single- and multichannel audio recordings database (SMARD)," in *Proc. Intl. Workshop Acoust. Signal Enhancement (IWAENC)*, Juan les Pins, France, September 2014, pp. 40–44.

[77] A. Blanco, M. Delgado, and M. Pegalajar, "A genetic algorithm to obtain the optimal recurrent neural network," *International Journal of Approximate Reasoning*, vol. 23, no. 1, pp. 67 – 83, 2000.

[78] M. Mitchell, *An introduction to genetic algorithms*, MIT press, 1998.

[79] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artificial Intelligence*, vol. 97, no. 12, pp. 245–271, 1997.

[80] M. A. Hall and G. Holmes, "Benchmarking attribute selection techniques for discrete class data mining," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 6, pp. 1437–1447, 2003.

[81] M. Robnik-Šikonja and I. Kononenko, "An adaptation of Relief for attribute estimation in regression," in *Machine Learning: Proceedings of the Fourteenth International Conference (ICML)*, Nashville, USA, 1997, pp. 296–304.

[82] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial intelligence*, vol. 97, no. 1, pp. 273–324, 1997.

[83] J. Bradley, R. Reich, and S. Norcross, "A just noticeable difference in C_{50} for speech," *Applied Acoustics*, vol. 58, no. 2, pp. 99–108, 1999.

[84] A. Khosravi, S. Nahavandi, D. Creighton, and A. F. Atiya, "Comprehensive review of neural network-based prediction intervals and new advances," *IEEE Transactions on Neural Networks*, vol. 22, no. 9, pp. 1341–1356, 2011.

[85] G. Papadopoulos, P. J. Edwards, and A. F. Murray, "Confidence estimation methods for neural networks: A practical comparison," *IEEE Transactions on Neural Networks*, vol. 12, no. 6, pp. 1278–1287, 2001.

[86] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Prentice Hall, Englewood Cliffs, NJ, 1978.

[87] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[88] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics bulletin*, vol. 1, no. 6, pp. 80–83, 1945.

[89] L. Sachs, *Applied statistics: a handbook of techniques*, Springer Science & Business Media, 2012.

[90] Y. Hochberg, "A sharper bonferroni procedure for multiple tests of significance," *Biometrika*, vol. 75, no. 4, pp. 800–802, 1988.

[91] A. M. Zoubir, V. Koivunen, Y. Chakhchoukh, and M. Muma, "Robust estimation in signal processing: A tutorial-style treatment of fundamental concepts," *IEEE Signal Processing Magazine*, vol. 29, no. 4, pp. 61–80, Jul 2012.

[92] J. Eaton, N. D. Gaubitch, A. H. Moore, and P. A. Naylor, "Estimation of room acoustic parameters: The ACE challenge," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 10, pp. 1681–1693, 2016.

[93] B.-H. Juang and L. Rabiner, "Automatic speech recognition: History," in *Encyclopedia of Language & Linguistics*, K. Brown, Ed., pp. 806–819. Elsevier, Oxford, second edition, 2006.

[94] X. Huang, A. Acero, and H.-W. Hon, *Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*, Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 2001.

[95] D. Yu and L. Deng, *Automatic Speech Recognition - A Deep Learning Approach*, Springer, London, October 2014.

[96] D. Jurafsky and J. H. Martin, *Speech and language processing*, Prentice Hall, London, second edition, 2008.

[97] P. A. Naylor and N. D. Gaubitch, "Speech dereverberation," in *Proc. Intl. Workshop Acoust. Echo Noise Control (IWAENC)*, 2005.

[98] R. Haeb-Umbach and A. Krueger, *Reverberant Speech Recognition*, pp. 251–281, John Wiley & Sons, Chichester, 2012.

[99] K. Kinoshita, M. Delcroix, S. Gannot, E. Habets, R. Haeb-Umbach, W. Kellermann, V. Leutnant, R. Maas, T. Nakatani, B. Raj, A. Sehr, and T. Yoshioka, "A summary

of the REVERB Challenge: state-of-the-art and remaining challenges in reverberant speech processing research," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, pp. 1–19, January 2016.

[100] M. Wölfel and J. McDonough, *Distant Speech Recognition*, John Wiley & Sons, Chichester, 2009.

[101] W. Li, L. Wang, F. Zhou, and Q. Liao, "Joint sparse representation based cepstral-domain dereverberation for distant-talking speech recognition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7117–7120.

[102] T. Yoshioka and T. Nakatani, "Noise model transfer using affine transformation with application to large vocabulary reverberant speech recognition," in *Proc. Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7058–7062.

[103] Y. Tachioka, S. Watanabe, and J. Hershey, "Effectiveness of discriminative training and feature transformation for reverberated and noisy speech," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 6935–6939.

[104] A. Schwarz, C. Huemmer, R. Maas, and W. Kellermann, "Spatial diffuseness features for DNN-based speech recognition in noisy and reverberant environments," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, Australia, April 2015, pp. 4380–4384.

[105] T. Yoshioka and M. Gales, "Environmentally robust ASR front-end for deep neural network acoustic models," *Computer Speech & Language*, vol. 31, no. 1, pp. 6586, 2015.

[106] A. Sehr, R. Maas, and W. Kellermann, "Model-based dereverberation in the log-melspec domain for robust distant-talking speech recognition," in *Proc. IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, 2010, pp. 4298–4301.

[107] L. Couvreur and C. Couvreur, "Blind model selection for automatic speech recognition in reverberant environments," *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 36, no. 2-3, pp. 189–203, 2004.

[108] M. Matassoni, A. Brutti, and P. Svaizer, "Acoustic modeling based on early-to-late reverberation ratio for robust asr," in *Acoustic Signal Enhancement (IWAENC), 2014 14th International Workshop on*, September 2014, pp. 263–267.

[109] L. Wang, Y. Kishi, and A. Kai, "Distant speaker recognition based on the automatic selection of reverberant environments using GMMs," in *Chinese Conference on Pattern Recognition (CCPR)*, November 2009, pp. 1–5.

[110] S. M. Ban and H. S. Kim, "Instantaneous model adaptation method for reverberant speech recognition," *Electronics Letters*, vol. 51, no. 6, pp. 528–530, March 2015.

[111] K. Kondo, Y. Takahashi, T. Komatsu, T. Nishino, and K. Takeda, "Computationally efficient single channel dereverberation based on complementary Wiener filter," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7452–7456.

[112] T. Ishii, H. Komiyama, T. Shinozaki, Y. Horiuchi, and S. Kuroiwa, "Reverberant speech recognition based on denoising autoencoder," in *Proc. INTERSPEECH*, 2013, pp. 3512–3516.

[113] B. W. Gillespie, H. S. Malvar, and D. A. F. Florêncio, "Speech dereverberation via maximum-kurtosis subband adaptive filtering," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2001, vol. 6, pp. 3701–3704.

[114] X. Xiao, S. Zhao, D. H. Ha Nguyen, X. Zhong, D. L. Jones, E. S. Chng, and H. Li, "Speech dereverberation for enhancement and recognition using dynamic features constrained deep neural networks and feature adaptation," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, pp. 1–18, 2016.

[115] M. Delcroix, T. Yoshioka, A. Ogawa, Y. Kubo, M. Fujimoto, N. Ito, K. Kinoshita, M. Espi, T. Hori, T. Nakatani, et al., "Linear prediction-based dereverberation with advanced speech enhancement and recognition technologies for the REVERB Challenge," in *Proc. REVERB Challenge*, Florence, Italy, 2014.

[116] T. Yoshioka, N. Ito, M. Delcroix, A. Ogawa, K. Kinoshita, M. Fujimoto, C. Yu, W. J. Fabian, M. Espi, T. Higuchi, S. Araki, and T. Nakatani, "The NTT CHiME-3 system: Advances in speech enhancement and recognition for mobile multi-microphone devices," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, December 2015, pp. 436–443.

[117] A. Sehr, R. Maas, and W. Kellermann, "Frame-wise HMM adaptation using state-dependent reverberation estimates," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 5484–5487.

[118] E. A. P. Habets, N. D. Gaubitch, and P. A. Naylor, "Temporal selective dereverberation of noisy speech using one microphone," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Las Vegas, USA, April 2008.

[119] M. L. Seltzer and R. M. Stern, "Subband likelihood-maximizing beamforming for speech recognition in reverberant environments," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, pp. 2109–2121, 2006.

[120] N. D. Gaubitch and P. A. Naylor, "Spatiotemporal averaging method for enhancement of reverberant speech," in *Proc. IEEE Intl. Conf. Digital Signal Processing (DSP)*, Cardiff, UK, July 2007.

[121] B. Kingsbury and N. Morgan, "Recognizing reverberant speech with RASTA-PLP," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1997, vol. 2, pp. 1259–1262.

[122] R. P. Lippmann, "Speech perception by humans and machines," in *Proc. of the ESCA Workshop on the "Auditory Basis of Speech Perception"*, 1996, pp. 309–316.

[123] L. F. Lamel, R. H. Kassel, and S. Seneff, "Speech database development: Design and analysis of the acoustic-phonetic corpus," in *Speech Input/Output Assessment and Speech Databases*, 1989, vol. 2, pp. 161–170.

[124] V. Zue, S. Seneff, and J. Glass, "Speech database development at MIT: Timit and beyond," *Speech Communication*, vol. 9, no. 4, pp. 351 – 356, 1990.

[125] K.-F. Lee and H.-W. Hon, "Speaker-independent phone recognition using hidden markov models," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 11, pp. 1641–1648, 1989.

[126] S. Sandhu and O. Ghitza, "A comparative study of mel cepstra and EIH for phone classification under adverse conditions," in *Proc. IEEE International Conference on Acoustics, Speech and SignalProcessing (ICASSP)*, 1995, vol. 1, pp. 409–412.

[127] A. K. Halberstadt, *Heterogeneous acoustic measurements and multiple classifiers for speech recognition*, Ph.D. thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, November 1998.

[128] D. Kolossa and R. Haeb-Umbach, *Robust speech recognition of uncertain or missing data: theory and applications*, Springer Science & Business Media, 2011.

[129] M. Mohri, F. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," *Computer Speech & Language*, vol. 16, no. 1, pp. 69–88, 2002.

[130] N. Kumar and A. G. Andreou, "Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition," *Speech Communication*, vol. 26, no. 4, pp. 283 – 297, 1998.

[131] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals, "WSJCAMO: a British English speech corpus for large vocabulary continuous speech recognition," in *Proc. IEEE International Conference on Acoustics, Speech and SignalProcessing (ICASSP)*, 1995, vol. 1, pp. 81–84.

[132] D. Schmid, P. Thuene, D. Kolossa, and G. Enzner, "Dereverberation preprocessing and training data adjustments for robust speech recognition in reverberant environ-

ments," in *Proc. of Speech Communication; 10. ITG Symposium*, September 2012, pp. 1–4.

[133] D. Vijayasenan, F. Valente, and H. Bourlard, "An information theoretic combination of MFCC and TDOA features for speaker diarization," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 2, pp. 431–438, February 2011.

[134] D. Korchagin, "Audio spatio-temporal fingerprints for cloudless real-time hands-free diarization on mobile devices," in *Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)*, May 2011, pp. 25–30.

[135] T. Stafylakis and V. Katsouros, "A review of recent advances in speaker diarization with bayesian methods," in *Speech and Language Technologies*, I. Ipsic, Ed., chapter 11, pp. 217–240. INTECH Open Access Publisher, Rijeka, 2011.

[136] R. Sinha, S. E. Tranter, M. J. F. Gales, and P. C. Woodland, "The Cambridge University March 2005 speaker diarisation system," in *Proc. European Conf. on Speech Communication and Technology*, Lisbon, Portugal, September 2005, pp. 2437–2440.

[137] S. Tranter and D. Reynolds, "An overview of automatic speaker diarization systems," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, pp. 1557–1565, 2006.

[138] R. Milner, O. Saz, S. Deena, M. Doulaty, R. W. M. Ng, and T. Hain, "The 2015 sheffield system for longitudinal diarisation of broadcast media," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, December 2015, pp. 632–638.

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

[140] N. Evans, C. Fredouille, and J.-F. Bonastre, "Speaker diarization using unsupervised discriminant analysis of inter-channel delay features," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Taipei, Taiwan, April 2009, pp. 4061–4064.

[141] 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)*, Kyoto, Japan, March 2012, pp. 4189–4192.

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

[143] T. Hori, S. Araki, T. Yoshioka, M. Fujimoto, S. Watanabe, T. Oba, A. Ogawa, K. Otsuka, D. Mikami, K. Kinoshita, T. Nakatani, A. Nakamura, and J. Yamato, "Low-latency real-time meeting recognition and understanding using distant microphones and omni-directional camera," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 2, pp. 499–513, February 2012.

[144] X. Anguera, C. Wooters, and J. Hernando, "Acoustic beamforming for speaker diarization of meetings," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 7, pp. 2011–2022, September 2007.

[145] C. H. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.

[146] E. El Khoury, C. Sénac, and P. Joly, "Audiovisual diarization of people in video content," *Multimedia Tools and Applications*, vol. 68, no. 3, pp. 747–775, 2012.

[147] Y.-C. Lu and M. Cooke, "Binaural estimation of sound source distance via the direct-to-reverberant energy ratio for static and moving sources," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 7, pp. 1793–1805, September 2010.

[148] D. Vijayasenan, F. Valente, and H. Bourlard, "Multistream speaker diarization beyond two acoustic feature streams," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Dallas, USA, March 2010, pp. 4950–4953.

[149] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," in *The 37th annual Allerton Conference on Communication, Control and Computing*, Allerton, IL, USA, September 1999, pp. 368–377.

[150] N. Slonim and N. Tishby, "Agglomerative information bottleneck," in *Advances in Neural Information Processing Systems 12*, S. Solla, T. Leen, and K. Müller, Eds., pp. 617–623. MIT Press, 2000.

[151] D. Vijayasenan, F. Valente, and H. Bourlard, "An information theoretic approach to speaker diarization of meeting data," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 7, pp. 1382–1393, September 2009.

[152] D. M. Brookes, "VOICEBOX: A speech processing toolbox for MATLAB," http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html, 1997–2013.

[153] J. G. Fiscus, N. Radde, J. S. Garofolo, A. Le, J. Ajot, and C. Laprun, "The rich transcription 2005 spring meeting recognition evaluation," in *Machine Learning for Multimodal Interaction*, pp. 369–389. Springer, 2005.

[154] "Objective measurement of active speech level," *ITU-T Recommendation P.56*, March 1993.

[155] G. McLachlan and T. Krishnan, *The EM algorithm and extensions*, vol. 382, John Wiley & Sons, New York, 2007.

[156] B. W. Silverman, *Density estimation for statistics and data analysis*, vol. 26, CRC press, 1986.

[157] D. Chauveau and D. R. Hunter, "ECM and MM algorithms for normal mixtures with constrained parameters," working paper or preprint, August 2013.

[158] L. Scrucca, "GA: A package for genetic algorithms in R," *Journal of Statistical Software*, vol. 53, no. 4, pp. 1–37, 2013.

[159] K. P. Burnham and D. R. Anderson, *Model selection and multimodel inference: a practical information-theoretic approach*, Springer Science & Business Media, 2002.

[160] C. Mitchell and L. Jamieson, "Modeling duration in a hidden markov model with the exponential family," in *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, April 1993, vol. 2, pp. 331–334.

[161] C. R. Shelton and G. Ciardo, "Tutorial on structured continuous-time markov processes.," *Journal of Artificial Intelligence Research*, vol. 51, pp. 725–778, 2014.

[162] E. Zwyssig, S. Renals, and M. Lincoln, "Determining the number of speakers in a meeting using microphone array features," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2012, pp. 4765–4768.