An Information Theoretic Approach to Speaker Diarization of Meeting Recordings

THÈSE N^{O} 4888 (2010)

PRÉSENTÉE LE 6 DÉCEMBRE 2010
À LA FACULTÉ SCIENCES ET TECHNIQUES DE L'INGÉNIEUR
LABORATOIRE DE L'IDIAP
PROGRAMME DOCTORAL EN GÉNIE ÉLECTRIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Deepu VIJAYASENAN

acceptée sur proposition du jury:

Prof. P. Vandergheynst, président du jury
Prof. H. Bourlard, Dr F. Valente, directeurs de thèse
Prof. M. Ostendorf, rapporteur
Prof. S. Renals, rapporteur
Dr J.-M. Vesin, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2010

Résumé

Dans cette thèse, nous étudions, dans le cadre de la théorie de l'information, une méthode non paramétrique de segmentation et de regroupement en locuteurs (SRL) de données audios de réunions. Le problème est formulé en utilisant le principe du goulot d'étranglement de l'information (principe IB pour Information Bottleneck). Contrairement aux autres méthodes où la distance entre des segments de locuteurs est introduite arbitrairement, le principe IB recherche la partition qui maximise l'information mutuelle entre les observations et les variables appropriées au problème tout en minimisant la distorsion entre les observations. L'optimisation de la fonction objectif du principe IB conduit à choisir la divergence de Jensen-Shannon comme distance entre segments de parole.

Dans la première partie de cette thèse, nous étudions la SRL reposant sur le principe IB et utilisant les coefficients spectraux MFCC (Mel-Frequency Cepstral Coefficients) comme primitives d'entrée. Nous traitons les questions liées à la SRL reposant sur le principe IB telles que l'optimisation de la fonction objectif du principe IB ou les critères permettant de déduire le nombre de locuteurs. Nous évaluons le système proposé en le comparant au système de référence en utilisant le jeu de données NIST RT06 (Rich Transcription) pour la SRL. Notre système reposant sur le principe IB atteint un taux d'erreur locuteur (16,8%) similaire au système de référence HMM / GMM (17,0%). Ce système utilisant un regroupement non paramétrique, il effectue la SRL six fois plus rapidement qu'en temps réel, alors que la méthode de référence est trop lente pour être réalisée en temps réel.

La seconde partie de cette thèse propose un nouveau système de combinaison de primitives dans le cadre de la SRL reposant sur le principe IB. Les deux étapes que constituent le regroupement et l'alignement des locuteurs sont discutées. Contrairement aux systèmes actuels, la méthode proposée permet de combiner des primitives tout en évitant le calcul de la moyenne des scores de log-vraisemblance. Deux ensembles différents de primitives ont été considérés - (a) combinaison de primitives MFCC avec des primitives de différences de temps d'arrivée (TDOA) (b) combinaison de primitives de quatre espèces dont MFCC, TDOA, spectre de modulation et prédiction linéaire fréquencedomaine. Les expériences montrent que le système proposé génère 5% d'amélioration absolue par rapport au système de référence lorsque deux types de primitives sont combinés (a), et 7% en combinant les quatre types (b). L'augmentation de la complexité de l'algorithme du système reposant sur le principe IB est minime lorsque l'on augmente le nombre de primitives. Le système utilisant les quatre types de primitives en entrée fonctionne en temps réel et est dix fois plus rapide que le système reposant sur les GMM.

Mots-clés : Segmentation et regroupement en locuteurs, données de réunions, principe du goulot d'étranglement de l'information, combinaisons de primitives.

Abstract

In this thesis we investigate a non parametric approach to speaker diarization for meeting recordings based on an information theoretic framework. The problem is formulated using the Information Bottleneck (IB) principle. Unlike other approaches where the distance between speaker segments is arbitrarily introduced, the IB method seeks the partition that maximizes the mutual information between observations and variables relevant for the problem while minimizing the distortion between observations. The distance between speech segments is selected as the Jensen- Shannon divergence as it arises from the IB objective function optimization.

In the first part of the thesis, we explore IB based diarization with Mel frequency cepstral coefficients (MFCC) as input features. We study issues related to IB based speaker diarization such as optimizing the IB objective function, criteria for inferring the number of speakers. Furthermore, we benchmark the proposed system against a state-of-the-art system on the NIST RT06 (Rich Transcription) meeting data for speaker diarization. The IB based system achieves similar speaker error rates (16.8%) as compared to a baseline HMM/GMM system (17.0%). This approach being non parametric clustering, perform diarization six times faster than realtime while the baseline is slower than realtime.

The second part of thesis proposes a novel feature combination system in the context of IB diarization. Both speaker clustering and speaker realignment steps are discussed. In contrary to current systems, the proposed method avoids the feature combination by averaging log-likelihood scores. Two different sets of features were considered – (a) combination of MFCC features with time delay of arrival features (b) a four feature stream combination that combines MFCC, TDOA, modulation spectrum and frequency domain linear prediction. Experiments show that the proposed system achieve 5% absolute improvement over the baseline in case of two feature combination, and 7% in case of four feature combination. The increase in algorithm complexity of the IB system is minimal with more features. The system with four feature input performs in real time that is ten times faster than the GMM based system.

Keywords: Speaker diarization, meeting data, information bottleneck principle, multistream diarization.

Acknowledgements

Most people say that doing a thesis is a very long ride. However, I never got to the point of *"When is then going to be over"* in the past four years. I really should thank my two supervisors big time for the great guidance which made it possible. I should specially thank Fabio for his great patience and persitence. At the same time, I am also indebted to Hervé for ensuring good quality work and his remarkable help with technical writing. In addition, I am grateful for the kind of freedom I had.

Let me thank my parents Vijayasenan and Vimala and my wife Resmy for the unconditional love and support they have been providing. I guess that has been my greatest strength ever.

Let me atleast make a list of some people whom I should thank for their helps and company.

- SriG and Kalika for convincing me about doing a Ph.D.
- The senior researchers John, Mathew and Phil for their helps.
- The system/admin team, especially Nadine and Sylvie for all their support and Michel for making me speak some French.
- The set of people who made me feel at home - Sriram, Sam, Jamie, Lakshmi and Saheer.
- The set of people who made sure I have some extra-curricular activites. Starting with Guil, Ganga and Ferran it is a long list. Benjamin, Jagan, Laurent, Marco Petr, Sriram, Tamara, Venki, Vijay to name a few.
- Anindya, Babu, Dayra, Joel, Gokul, Hari, Remya, and my other Idiap colleagues for keeping me good company.

Last but not least, I should really thank a special set of my friends back home – Ajiyos, Arun, Don, Kannan, Randeep, Simy and Vincy.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The exponential increase in storage capacity and processing power has facilitated digitization of large volumes of spoken documents like broadcast, meetings, lecture room conversations. In this context, there is a strong need to apply automatic speech processing methods to achieve effective searching and indexing of those data. This requires other technologies besides automatic speech recognition. Speaker diarization is one of such technologies addressing the problem *"who spoke when"*.

In recent years, diarization has been applied to spontaneous multi-party discussions also informally known as meeting recordings. Speaker diarization is trivial if meeting participants have their own recording instruments such as lapel microphones or individual headset microphones. However, audio recording is often performed in a "non-intrusive manner" using Single Distant Microphone (SDM) or Multiple Distant Microphones (MDM).

MDM is generally a collection of one or more microphone arrays (circular or linear). the number of microphones can vary from two to up to sixty four microphones depending on the meeting room equipment.

Speaker diarization finds its application across a wide variety of domains. Some applications include:

- The speaker segmentation information can be used to refine the raw text output of Automatic Speech Recognition (ASR) into a conversational form that is useful in related tasks such as meeting summarization. The segmentation output is also used by speaker adaptation algorithms to improve the ASR performance.

- In case of large multimedia archives, obtaining speaker labels can be useful in terms of speaker indexing and retrieval across different documents.

- Most of the meeting analysis algorithms include speaker diarization as a first step. Speaker diarization output contains useful characteristics of meeting participants like total speaking length, average speaking duration etc. These features are employed in automatic analysis of interaction between participants, e.g., dominance detection, role recognition, identifying whether a meeting is co-operative or not.

1.1 Motivations

Speaker Diarization involves two simultaneous tasks: (1) the estimation of the number of speakers in an audio stream (2) associating each speech segment to each speaker. In recent years, diarization of meeting domain data has been an active research field.

Conventional methods perform speaker diarization based on parametric models such as Hidden Markov Model (HMM) with Gaussian Mixture Models (GMM) as the emission probabilities. Recent advances show that significant improvements can be obtained combining traditional short term spectral features, e.g., Mel Frequency Cepstral Coefficients (MFCC), together with the Time Delay of Arrival (TDOA) extracted from the microphone array as well as with other features extracted from long time spans of the signal.

While the combination several information sources is effective in reducing the diarization error, it introduces a number of weaknesses and challenges that are addressed in this work, i.e, the computational complexity of the system and the most effective and robust way of integrating multiple features.

- The conventional models based on the HMM/GMM framework estimate separate parametric models for each hypothesized speaker. This is computationally expensive and requires considerable optimization to achieve real time performance. Whenever multiple features are used, the computational complexity of the diarization system further increases.

  In the context of the meeting processing, there is a strong requirement to have diarization systems with low computational complexity. Such a speaker diarization system would enable applications such as meeting browsing, meeting indexing and retrieval on a normal desktop machine.

- The dimensionality of TDOA feature vector is variable, and the combination of MFCC and TDOA features often presents a robustness problem. The combination of a fixed dimension MFCC vector with a variable dimension TDOA vector does not always produce consistent improvements.

- Beside TDOA features, other combinations have been proven effective, e.g., MFCC plus prosodic features, MFCC plus modulation spectrum features. However no positive results have been reported to date on the combination of more then two feature streams.

1.2 Objective

The main objective of the thesis is to advance the field of speaker diarization of meeting data recorded with multiple distant microphones along two axes:

- The reduction the computational complexity of the system with little loss in performances so that real time diarization is possible even when a large number of features are integrated in the system.

- The inclusion of multiple features, beyond the common two streams paradigm (MFCC and TDOA), in order to further reduce the diarization error.

1.3 Contributions

Towards those objectives, this thesis investigate the use of a non-parametric clustering algorithm for speaker diarization. The method is based on the Information Bottleneck (IB) principle which operates on a space of “relevance variables”. The clustering method attempts to preserve maximum mutual information with respect to relevance variables.

The main contributions are:

- The first part of the thesis casts the speaker diarization problem using the Information Bottleneck method.

  Being non-parametric, the algorithm does not estimate explicit parametric models for each speaker. Hence the method is computationally less expensive than conventional HMM/GMM systems. The proposed system could achieve similar performance while being significantly faster [134].

- The second part of this thesis extends the IB framework to handle multiple features streams. Experiments reveal that the increase in computational complexity is significantly lower compared to the HMM/GMM case. Furthermore the combination is more robust to the different feature. In particular whenever the time delay of arrivals are used together with the MFCC features, the IB system is more robust to the dimensionality of the TDOA vector producing lower speaker error than the HMM/GMM system [138].

- The third part of the thesis investigates the use of additional features to the MFCC and TDOA combination. Two different sets of features based on long temporal context are explored. The combination of four feature streams results in significant improvements in case of IB diarization; in contrast to the marginal benefits in case of baseline HMM/GMM from this combination. As per our best knowledge, this is the first successful attempt in combining more than two features for speaker diarization [137].

- The time complexity of the proposed system with multiple input features does not increase considerably as compared to the conventional systems. The proposed diarization system with four input feature streams is still faster than real-time [136].

In summary this thesis proposes and investigates a novel diarization system based on the Information Bottleneck principle which allows the integration of a large number of features while still performing the task faster then real time.

1.4 Organization

The structure of this thesis is as follows:

- Chapter 2 is an overview of different algorithms that perform various steps in typical speaker diarization algorithms.
- Chapter 3 introduces single stream diarization system based on Information Bottleneck Principle. The relevance information is provided through a background GMM and the clustering is performed by agglomerative optimization. Two criteria for model selection (detecting the number of speakers) are investigated.
- Chapter 4 extends the diarization system to incorporate time delay of arrival features. The chapter describes the new distribution based combination of feature streams. Subsequently, a realignment algorithm based on IB principle is described. The algorithm is evaluated using MFCC and TDOA input features.
- Chapter 5 further investigates an extension of the two stream system with two additional acoustic features with long temporal context. Different issues such as algorithms robustness are investigated in comparison with a traditional HMM/GMM system for the multi-stream combination.
- Chapter 6 reports the results of the algorithms on recent NIST evaluation datasets.
- Chapter 7 provides a short summary and conclusions of the thesis.

Chapter 2

Speaker Diarization: An Overview

The goal of speaker diarization is to segment a continuous audio recording into different segments and to annotate each segment with a "speaker label" as shown in Figure 2.1. This includes determining how many speakers are present in the meeting as well as identifying the speech segments of each speaker in an unsupervised manner.

In this chapter, we review key sub-tasks needed to perform speaker diarization namely,

- **Feature Extraction:** This step extracts the speaker characteristics from the audio (Section 2.1).

- **Speech Activity Detection (SAD):** This is the task of speech non speech separation (Section 2.2).

- **Speaker Change Point Detection** This step divides the input speech into short segments that contain only one speaker (Section 2.3).

- **Speaker Clustering:** This step operates on the output of change point algorithm and groups

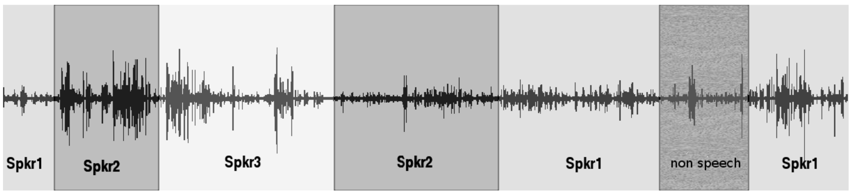


Figure 2.1. Speaker diarization consists of detecting number of speakers and speech corresponding to each speaker

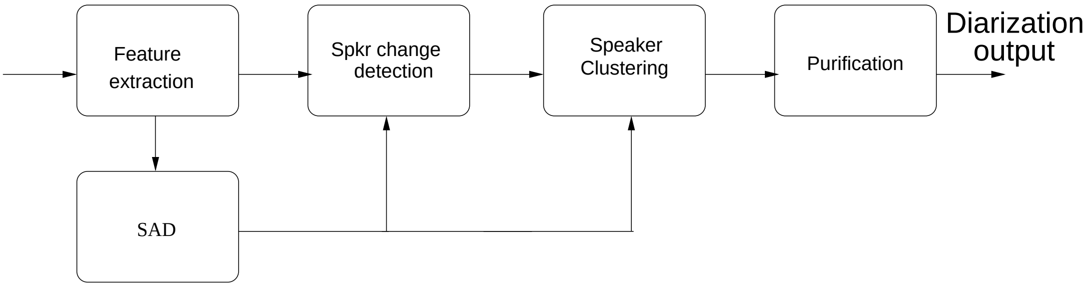


Figure 2.2. Various stages in speaker diarization

the speech segments of same speaker together (Section 2.4).

The block diagram of a typical diarization system is illustrated in Figure 2.2. We examine different approaches to each of these blocks in the following.

2.1 Feature Extraction

The objective of feature extraction is to extract the information that is useful for the task and to suppress all irrelevant information. Speaker diarization systems generally use one or more of the following features:

- **Short term spectrum** based features are extracted from the Fourier transform of speech segments considered in short analysis time windows. This is discussed in Section 2.1.1.
- **Time delay of Arrival (TDOA)** features are extracted from audio recorded with multiple microphones. These features contain speaker location information and will be described in Section 2.1.2.
- **Other features** include prosodic features such as pitch and modulation spectrogram and will be described in Section 2.1.3.

2.1.1 Short term spectral features

Most of the speaker diarization systems employ features based on short term spectrum of the signal. The input audio is windowed using a hamming window of around 25ms duration with a 50% overlap of successive windows. The spectral features are then extracted from the short term Fourier transform of the windowed signal. The most common features used in this context are the Mel frequency Cepstral coefficients (MFCC) (eg: [2; 5]). Speaker diarization systems use a higher number (around 20) of cepstral coefficients as compared to Automatic Speech Recognition systems. Unlike ASR systems, the delta and delta-delta coefficients are not employed widely since they capture short-time variations of the signal (tens of milliseconds) that generally capture phoneme evolution. Only a few attempts make use of delta features for diarization [150; 71]. Other short term features explored in literature include Line Spectrum Pair (LSP) frequency features [1], Perceptual Linear Prediction (PLP) coefficients [104] and Linear Predictive Cepstral Coefficients [3].

In order to reduce the effect to background noise and other unrelated acoustic events, feature warping techniques [91] are proposed to gaussianize the pdf of the features prior to modeling. Significant improvements are reported [104; 152]. In the same context, RASTA-PLP features [49] are used in AMI speaker diarization system [128; 129].

2.1.2 Time Delay of Arrival Features

In case of meeting room data with multiple microphones, the time delay of arrival (TDOA) of signals in different microphones was found to be useful [65; 89]. The TDOA features are calculated by finding the peak of the cross correlation between channels and can be estimated without explicit knowledge of the microphone array geometry. Another set of features was proposed in the context of multiple microphones in [86]. The feature set consists of a subspace approximation of energy ratios in different sub-bands. These features were found to be more robust to overlapped speech [87]. In another attempt, a discriminant analysis with respect to an initial set of clusters is performed over TDOA features [31]. The initial clustering was made with a threshold on the Euclidean distance between features. The method reported consistent improvement over meetings that have more than 2 microphone inputs. The most common approach is to use TDOA features in combination with MFCC features. This achieves the state of the art in speaker diarization in different speaker diarization evaluations [10; 143; 114; 115].

2.1.3 Other Features

Long term features consider the speech in larger segments which reveals long term speaker characteristics, that are not captured by the short term analysis. Features such as pitch frequency, energy, centroid of peak frequency and peak frequency bandwidth are examples of features considered for speaker segmentation [148]. In addition, three new features based on cross correlation of the signal power spectrum – temporal feature stability, spectral shape and white noise similarities – are investigated. A wide variety of prosodic features are presented in [35] in terms of their discriminability for speaker diarization.

To complement the short-time spectral features, modulation spectrogram features that depend on a longer temporal context were employed in [139]. In a similar work, the combination of MFCC and a set of prosodic and long term features such as energy based features, harmonic to noise ratio, long term average spectrum were found to improve the diarization performance [35]. In a related work [51], the prosodic features were used to initialize the clusters for agglomerative clustering.

2.2 Speech Activity Detection

Speech activity detection(SAD) refers to identification of regions that contain speech from the participants present in the meeting recording. The non-speech regions may contain silence, any meeting room noise, other sounds such as laugh, background music etc. A speech/non speech separation is essential prior to any speaker segmentation or clustering.

Speech/non-speech detectors can be generally classified into different categories:

1. **Energy based detectors** perform the speech non speech classification based on short term energy of the signal (Section 2.2.1).

2. **Model based speech-non speech detection** builds separate speech and non speech models from the input features (Section 2.2.2).

3. **Hybrid schemes** depend on energy based schemes to obtain an initial classification to train the speech and non speech models in an unsupervised manner (Section 2.2.3).

4. **Multichannel speech activity detection** attempts to perform speech/non-speech separation from a multichannel audio recording (Section 2.2.4).

2.2.1 Energy Based Detectors

Energy based speech detection works based on thresholding the short term energy of the signal. This scheme is simple and require no training data and is used in telephone audio in general. The algorithms perform noise adaptation to track the non-stationarity of the signal [61; 59]. However, meeting recordings contain various high energy non speech events such as laughing. In this case the speech detection performance with energy based systems degrades as the non-speech events also have high energy. In addition, speech acquired using distance microphones has relatively low SNR. This degrades the performance of energy based systems for meeting recordings [52; 127].

2.2.2 Model Based Detectors

Model based detectors are very common for speech activity detection for meetings. They build multiple models for different characteristics of the audio from a development dataset. General approach consists of training Gaussian Mixture Models (GMMs) using labeled data. A Viterbi decoding on the unsegmented audio using these models provides the required speech/non speech segmentation. The models are used in segmenting the input audio to speech and non-speech region. The simplest system uses only two models for speech and noise [144].

In case of broadcast data, multiple speech models are trained for different gender and bandwidth combinations [79] or to model noise and music separately [39; 151]. Those works also model classes like speech+noise and speech+music that helps in reducing rejection of speech frames as non speech. This data is subsequently classified as speech.

One advantage with this approach is that it can deal with very specific speech / non-speech classes. However, the method needs labeled training data that requires manual annotation. In addition, with complex models the labelled data may not be sufficient to train the parameters and may not generalize across different recording conditions.

An alternative approach consists of employing multi layer perceptron (MLP) for speech/no-speech separation [30] in meeting recordings. The MLP parameters are estimated from data that is labelled with force alignment. A Viterbi decoding with scaled likelihoods that are generated from the MLP classifier perform the speech non-speech segmentation.

2.2.3 Hybrid Schemes

Hybrid systems attempt to remove the data dependency in case of model based approaches. They generally consist of two stages – an energy based first step followed by a model based detector. The output of the energy based detector can be used as training data for the second stage and thus the dependency on labeled training data is eliminated. A derivative filter often precedes the energy thresholding to enhance the change points [8; 10]. The second stage is a model based detector that is typically based on GMM models. The system parameters are estimated using segmentation from the first stage. Multiple iterations of training and segmentations are performed using an EM algorithm. As in case of model detectors, multiple models for non speech such as silence, non-speech sound models are explored [50; 143].

2.2.4 Multichannel detectors

Recent research efforts concentrate on multichannel audio recordings. In this context there have been attempts to address multi-speaker speech activity detection. An important challenge in this domain is to avoid cross talk. Features for classifying the multichannel audio into four classes namely local channel speech, local channel speech with cross talk, cross talk and silence was investigated in this context [145]. Another method to discard cross talk speech segments is based on a post processing step that thresholds cross correlation between channels [92]. An alternate algorithm detects cross-talk by means of peak of joint cross correlation [64]. This method did not require any training data and was later extended to a hybrid system using Gaussian models for detection [63].

2.3 Speaker Change Detection

Speaker Change Detection algorithms divide the output of the SAD system into segments that contain speech of only one speaker. The algorithm tries to detect all speaker change points to form “speaker-pure” segments so that they can be clustered in the next step. Common approach to change detection involves computation of a distance measure between two segments of speech data X_i and X_j around the hypothesized change point. The distance is then compared with a threshold to determine if a speaker change occurs. The threshold for the decision is empirically determined based on experiments conducted in development data. Various speaker change detection algorithms are proposed in the literature based on the distance measure and the choice of decision threshold. The most common distance measure is based on Bayesian Information Criterion. However, various other forms of distance measures such as symmetric KL divergence, Gish distance, Generalized Log likelihood Ratio (GLR) exist in literature.

2.3.1 KL based change detection

KL divergence measures dissimilarity between two distributions. In case of speaker change detection, the adjacent windows are declared as belonging to same speaker if the KL divergence is below an empirically determined threshold. Most of the speaker segmentation algorithms based on KL is based on a closed form expression of KL divergence in case of Gaussian distributions.

$$KL(X_i||X_j) = \tfrac{1}{2}tr[(\Sigma_i - \Sigma_j)(\Sigma_j^{-1} - \Sigma_i^{-1}) + (\Sigma_j^{-1} - \Sigma_i^{-1})(m_i - m_j)(m_i - m_j)'] \tag{2.1}$$

where m_l, Σ_l are the mean and covariance of data X_l. A symmetric form of KL divergence is given by the sum of two KL divergences:

$$KL2(X_i, X_j) = KL(X_i||X_j) + KL(X_j||X_i) \tag{2.2}$$

This symmetric KL divergence is used as the distance measure where models of the data in the two segments in consideration are Gaussian distributions [103]. It was observed that the environment conditions introduce a bias in mean estimates and a refined measure known as divergence shape distance was proposed to avoid the dependency on mean estimates [60].

Since no exact closed form expression exists for mixture models, use of KL divergence is mostly limited to Gaussian speaker models that have limited capabilities in modeling. However, KL divergence is often used as the first stage [28; 153] to decrease the computational cost.

2.3.2 Generalized Likelihood Ratio Test

Earliest approaches to speaker change detection scheme rely on Generalized Likelihood Ratio (GLR) test. The generalized ratio test [141] was adopted to segment the speaker turns with a tuned threshold[20; 36; 1]. Speaker segmentation is based on whether the two adjacent windows X_i and X_j of speech is modeled with a single gaussian $\mathcal{N}_{i+j}$ trained on the entire data $X_i \cup X_j$ (no speaker change) or with separate gaussians $\mathcal{N}_i$ and $\mathcal{N}_j$ (speaker change detected). The distance measure is computed from the likelihoods as:

$$D_{GLR} = -log\left[\frac{\mathcal{L}(X_i \cup X_j|\mathcal{N}_{i+j})}{\mathcal{L}(X_i|\mathcal{N}_i)\mathcal{L}(X_j|\mathcal{N}_j)}\right] \tag{2.3}$$

N_{i+j} denotes the Gaussian distribution estimated from data X_i and X_j combined.

Gish distance measure is derived from GLR [40; 41] avoiding the dependency on mean estimates as in case of divergence shape distance and is expressed in closed form as:

$$D_{Gish}(X_i, X_j) = -\frac{N}{2}\log\frac{|\Sigma_i|^{\alpha}|\Sigma_j|^{1-\alpha}}{|\alpha\Sigma_i + (1-\alpha)\Sigma_j|} \tag{2.4}$$

where Σ_i, Σ_j are the covariance matrices for X_i and X_j and $\alpha = N_i/(N_i + N_j)$.

Though GLR can be computed for the mixture models, the measure does not take the complexity of the models into consideration. The log-likelihoods can be arbitrarily increased with introducing more mixture components. This issue is addressed by Bayesian Information Criterion(BIC) that try to penalize models with high complexity [24] which is described below.

2.3.3 Bayesian Information Criterion

Bayesian Information Criterion(BIC) was originally introduced in the case of model selection [101] and was later applied to speaker change detection [24]. For an acoustic segment X_i BIC value of a model M_i is given by:

$$BIC(M_i) = \log\mathcal{L}(X_i|M_i) - \frac{\lambda}{2}\#(M_i)\log(N_i) \tag{2.5}$$

where N_i denotes number of samples in X_i and $\#(M_i)$ denotes the number of parameters for the model. λ is a tunable parameter that controls the trade-off between model complexity (number of parameters) and the best fit of the data (large likelihood). Speaker segmentation is based on whether the two windows X_i and X_j in question are modeled with a single distribution, M_{i+j} trained on entire data $X_i \cup X_j$ (no speaker change) or each with separate distributions M_i and

M_j (speaker change detected). i.e., The two speech segments X_i and X_j are compared based on difference between the BIC value of the combined window and the sum of BIC values of the two individual segments and is given by:

$$\Delta BIC_{ij} = \log \mathcal{L}(X_i \cup X_j | M_{i+j}) - [\log \mathcal{L}(X_i|M_i) + \log \mathcal{L}(X_j|M_j)] - \frac{\lambda}{2}\#(\delta_{ij}) \log N \tag{2.6}$$

where $\#(\delta_{ij}) = \#(M_{i+j}) - [\#(M_i) + \#(M_j)]$ denotes the change in number of parameters. Whenever a change point is detected, the window boundaries are restarted from the change point again. In case no speaker change is observed, the window size is increased and the process repeats. The BIC criterion in this form contains a tunable parameter λ in the penalty term that implicitly defines the threshold for comparison. Further studies have investigated different methods to improve this scheme. An alternate penalty selection was found to improve the method [121]. Another scheme eliminates the use of this threshold by adjusting the number of parameters between models [4]. In this scheme, the number of model parameters is kept constant before and after merge. i.e., the number of components of the model M_{i+j} is equal to the sum of the number of components in M_i and M_j. Thus the last term in Equation 2.6 reduces to zero, thus eliminating the tunable parameter λ.

BIC is still the state of the art for many different applications. However, two major issues are reported with BIC systems [119] and are given by:

- BIC has high miss rates on detecting short turns ($< 2 - 5s$), that can be problematic to use on fast interchange speech like conversations.

- the full search implementation is computationally expensive.

There has been considerable research to overcome these issues for BIC. While one approach [121] tries to consider a variable window where the size of the window is increased adaptively, an alternate scheme tries to avoid some unlikely BIC computations in order to decrease computations. A MAP adaptation of speaker models was presented [95] that is able to detect speaker change points in a short duration.

In order to reduce the computational complexity many systems apply a computationally less expensive distance measure as an approximation to BIC in the first stage. Only the change points detected in this stage are passed to the BIC computation. DIST-BIC [29; 28] is such an algorithm where a Log likelihood ratio test or KL divergence is used in the first stage. This work was further extended to refine speaker change position location and change candidate detection [153]. Another approach based on applying T^2 statistics prior to BIC [149] which when combined with variable window length increasing schemes resulted in close to 100 times improvement in the performance time. A similar approach employs a symmetrical KL divergence measure on Line Spectrum Pair(LSP) frequency features and the system is able to meet realtime processing requirement [67].

Alternate distance measures to reduce the computational complexity include normalized log likelihood ratio (NLLR) [130] and cross BIC (XBIC) [7]. XBIC distance measure is shown to be faster than BIC, while having similar performances. A normalized version of XBIC was found to be more robust against speaker variations [68].

Other approaches towards reduction of computational complexity include employing a variable window length is used to reduce the number of computations [105] and a divide and conquer approach where the long segments are segmented first and then further proceeds towards shorter time segments[25]. In the latter scenario, both stages use BIC criterion and the method reports significant improvements over single stage BIC.

2.4 Speaker Clustering

Speaker clustering combines segments from same speaker together. Ideally one cluster is produced for each speaker and all speech segments from a given speaker are assigned to the same cluster. Most of the state-of-the art systems follow a top-down or bottom-up hierarchical clustering. The clusters of speech segments are iteratively split (agglomerative/top-down) or merged (divisive/bottom-up) until a stopping criteria is reached. Top down systems start with few large clusters and divide them until the optimum number of clusters. In contrast, bottom-up methods over-segment the speech segments into a large number of clusters which are merged until the optimum number of clusters. Either case would require two quantities to be defined:

- **a distance measure** that defines the speaker similarity between two speech segments/clusters

- **a stopping criterion** to terminate the merging / splitting at the optimal number of clusters

```
Input:
    Speech segments $x_1, \ldots, x_N$ from speaker change detection/uniform linear segmentation
    Stopping Criterion
    Distance measure $d(.,.)$
Output:
    $C_m$: $m$-partition of $X$, $1 \leq m \leq N$
Initialization:
  • Initial partition $c_i = x_i, i = 1, \ldots, N$
Main Loop:
    While stopping criterion != true
        merge $c_i, c_j$ such that $d(c_i, c_j)$ is minimum
```

Figure 2.3. Agglomerative Speaker Clustering

Most of the research focuses on definition of these quantities that is discussed below.

2.4.1 Agglomerative Approaches

Agglomerative or bottom-up clustering is the most common approach towards speaker clustering since it could incorporate the output of speaker change detection algorithms as the starting point. Normally a distance metric between all clusters is computed at each stage and the closest pair is merged until the stopping criterion is satisfied. Figure 2.3 presents the pseudo code.

Many distance measures that are used in case of speaker change detection can be applied for agglomerative speaker clustering. The earliest works in this domain depend on modeling speakers with Gaussian distributions with Gish distance or symmetric KL divergence as the distance measure [54; 103; 149]. The most common stopping criterion is based on an empirical threshold on the minimum distance between pairs. Minimization of a penalized version of within cluster dispersion was also explored to determine the optimal number of clusters in case of Gish distance [54]. In this work, the Gish distance is weighted to favor merging of neighboring segments. The Gaussian model might be too restrict to characterize speaker properties in this case.

State of the art algorithms use Gaussian mixture modeling of speakers and Bayesian Information Criterion (BIC) is again the most common distance measure as in the case of speaker change detection. Starting from each segment as a speaker cluster, hierarchical clustering is performed. The BIC measure is calculated for every cluster pair, and the one with highest BIC is merged together. Though the clustering uses a threshold that is based on model selection framework, experiments on broadcast news [121; 130; 120] revealed that tuning this threshold is essential not only to improve performance but also to generalize well on unseen data. A similar scheme was employed for meeting diarization also [127]. A significant contribution in this domain eliminates the penalty term by fixing the number of parameters the same before and after merge step [4]. BIC is however a computationally expensive algorithm since each distance calculation involve a GMM estimation step. Different strategies for fast computation of BIC was explored in literature [121; 130].

Many speaker clustering algorithms use speaker models that are adapted from a common background GMM, that is estimated from the speech data from the same meeting. In the special case when all speaker model GMMs are mean adapted from the background GMM based on MAP adaptation, an upper bound on KL divergence between two Gaussian models was used as the distance measure [17]. The method adopts a threshold on the distance measure as the stopping criterion and outperforms BIC in a broadcast news corpus. However, the method was found to be less robust compared to BIC with errors in segmentation and was improved by replacing the stopping criterion with a BIC criterion [76]. A similar distance measure based on KL divergence between individual Gaussian distributions of a GMM allows fast distance computation as well as updation of speaker models [96]. Another choice of distance measure in the case of GMM speaker models is based on Generalized Likelihood Ratio(GLR) test [112; 55]. The algorithm use BIC as the stopping criterion. To force the merge of clusters that are close in time, a penalized GLR was proposed [140].

Motivated by speaker recognition systems, many speaker clustering algorithms use a Universal Background Model (UBM). The UBM is estimated from a large number of speakers from a separate training dataset. The speaker models are estimated by performing adaptation of the UBM model for individual speaker segments. In this framework, Cross likelihood distance is used to perform agglomerative clustering, which is initialized with a BIC based clustering with an early stopping (more number of clusters than required) [15; 151; 152; 104]. The cluster models are then compared using cross-likelihood distance for further merging where an empirical set of thresholds define the stopping criteria. The cross likelihood distance is given by:

$$D(X_i, X_j) = \frac{1}{N_i}\frac{\mathcal{L}(X_i|M_{UBM+j})}{\mathcal{L}(X_i|M_{UBM})} + \frac{1}{N_j}\frac{\mathcal{L}(X_j|M_{UBM+i})}{\mathcal{L}(X_j|M_{UBM})} \tag{2.7}$$

where M_{UBM} denotes the background model and M_{UBM+j} denotes the MAP adapted of UBM models with data X_j. Another related work proposes a Common Variance GMM (CVGMM) which is a tied covariance model to take into account small duration clusters better[83]. A model selection (GMM/CVGMM) is performed by a BIC criterion. Another novel distance metric is based on a technique called triangulation [73]. Given a set of acoustic segments $\{X_k\}$ and a set of initial clusters $\{C_i\}$, each cluster is represented in terms of the conditional likelihoods $p(C_i|X_k)$. The distance measure between clusters is computed in terms of cross correlation between such vectors:

$$D(C_i, C_j) = \sum_k p(C_i|X_k)p(C_j|X_k) \tag{2.8}$$

The method can be seen as a projection into a speaker space followed by the computation of the distance measure.

An alternative distance metric, namely relative entropy distance [58] is used in speaker clustering [98] to improve speech recognition performance. The relative entropy distance between two segments X_i and X_j with GMM speaker models M_i and M_j is:

$$D(i, j) = \frac{1}{2}\left[D_{\lambda_i\lambda_j} + D_{\lambda_j\lambda_i}\right] \tag{2.9}$$

$$D_{\lambda_k\lambda_l} = \mathcal{L}(X_k|M_k) + \mathcal{L}(X_k|M_l) \tag{2.10}$$

The stopping criterion is again based on a threshold on the distance measure. The same framework is also utilized for other datasets [48]. The framework was later refined [99] to train a single GMM on all speech segments with a separate mixture weight distribution adapted to each segment. The distance between segments is defined in terms of relative increase in entropy of this weight distribution due to clustering of two segments.

Another approach for speaker clustering depends on Fisher voices [26]. A background GMM is trained on the meeting whose means are then MAP-adapted to each speaker segment. The means of the adapted GMMs are considered to form a super-vector. The set of super-vectors are projected to a discriminative subspace(estimated from a development dataset) and agglomerative clustering is performed with Euclidean distance metric.

2.4.2 Divisive Clustering

Divisive or top-down approaches are not very common as speaker clustering algorithms. A top down split and merge clustering framework [57] splits data into upto four sub-clusters and allows for merging of similar sub-clusters. Two implementations are proposed for the algorithm – the first one based on MLLR adaptation and the other one based on Arithmetic Harmony Sphericity (AHS) metric [18] – to assign speech segments to the sub-clusters. The AHS distance measure is defined as:

$$AHC(X_i, X_j) = \log[tr(\Sigma_i \Sigma_j^{-1}).tr(\Sigma_j \Sigma_i^{-1})] - 2\log(d) \tag{2.11}$$

where Σ_l is the covariance matrix of speech segment X_l and d is the feature dimension. The splitting is terminated based on a minimum occupancy criterion. This algorithm was further used for speaker diarization task [56; 120].

2.4.3 Other Approaches

There are set of algorithms that do not belong to agglomerative/divisive framework for speaker clustering.A typical example is a speaker clustering algorithm based on Self Organizing Maps (SOM) [62]. It is a VQ clustering algorithm that generate the code books for each speaker. The optimum number of clusters is defined by applying BIC on a likelihood function defined over the code vectors of the codebook.

Another approach maximizes the within-class homogeneity to optimize the speaker clustering [122]. The within-cluster homogeneity is characterized by the likelihood probability that a cluster model, trained using all the utterances within a cluster, matches each of the within-cluster utterances. The maximization is performed either by a genetic algorithm or an alternative solution based on a divergence based model similarity. Model selection is performed using BIC.

Spectral clustering approaches were also successfully employed towards speaker diarization [80]. Individual speech segments are modeled with GMM and an approximation of KL divergence is used as a distance measure. Spectral clustering [78] is performed using an affinity matrix constructed based on this distance measure. Two model selection algorithms based on the eigen decomposition of the affinity matrix are explored.

Variational Bayesian(VB) learning was explored in the context of speaker clustering [124; 125] where both speaker models and the complexity of the model are learned at the same time. The system can be initialized with high number of clusters and gaussians per speaker, and the system would eliminate the cluster models and Gaussian components that are not used. This work was later extended in the context of meeting diarization [126].

This work was later extended to consider the use of infinite models for speaker clustering[123]. Speaker segmentation is obtained through a Dirichlet Process Mixture model and learning is based on a VB approximation. This method shows improvements over ML/BIC and MAP/BIC

2.4.4 System Combination

There were only a few attempts to combine the output of multiple speaker clustering algorithms. A cluster voting scheme was designed to reduce the diarization error by combining the results from two different diarization systems[118]. Two different sets of systems are tested with the combination and consistent improvements with respect to the individual performances are reported. Other approaches to combine information from different diarization systems are hybridization and merging [74; 75]. The hybridization method uses the segmentation result of a bottom-up system to initialize a top-down system. Merging strategy proposes associating the common result segments followed by a re-segmentation of the data to assign the segments where the two algorithms differ. A similar approach trains MAP-adapted GMMs from speech segments where two input systems agree and these adapted models are used for iterative segmentation [21].

2.4.5 Joint segmentation and clustering

One important class of algorithms perform a model-based speaker segmentation and clustering tasks jointly and eliminate the requirement of an initial speaker change detection algorithm. An iterative approach combines both segmentation and clustering in a framework referred as evolutive HMM (E-HMM)[70]. Initially the system starts with one HMM trained over all the acoustic data available. The best subset of features that gives maximum likelihood is selected and is used to create a MAP adapted HMM. A segmentation is performed on the data by Viterbi decoding. These steps are followed iteratively. A tunable parameter based on gain in the likelihood score is used as the stopping criterion. A similar approach introduces a repository model that further improve the purity of the clusters [6].

One of the popular approaches employed an ergodic HMM model [3] used for agglomerative clustering. The algorithm is initialized with uniform linear segmentation of input segments. Multiple iterations of re-estimation and segmentation are performed. Once the models converge, two nearest cluster pairs according to a modified Bayesian Information Criterion(BIC) are merged together. The stopping criterion is determined by the same BIC criterion. This framework was utilized by many systems in literature for speaker diarization. Alternatively Viterbi segmentation likelihoods are used in the stopping criterion [144]. Various improvements to this system include, a purification algorithm to split acoustically non-homogeneous clusters[11]; a new algorithm to initialize the input clustering [10]. These systems achieve the state of the art in NIST "Meeting Diarization Evaluation" task with a combination of MFCC and TDOA systems.

2.4.6 Purification of output clusters

The agglomerative approaches follow a greedy strategy that could result in a segmentation that is a local minimum. Often a post processing step further refines the clustering output.

A purification method involves the algorithm to first find the best speech segment (the segment with highest likelihood) for each cluster. Subsequently a ΔBIC value is computed between the best segment and every other segment in the same cluster. Depending on a threshold, either the model is declared pure or split into two clusters. In case of a split, all models are retrained and the data is re-segmented [9].

Another algorithm with the same goal was proposed where speaker models are re-estimated multiple times and the input data is re-segmented with the updated models [81]. In order to prevent over-fitting, each cluster is divided into two equal parts randomly. While the model estimation step performs re-estimation from one part, the segmentation step updates the cluster labels of the other part. In the next step the role of the two parts are reversed. The method is referred as "cross EM refinement".

2.5 Resources for Speaker diarization

This section presents a small overview of available datasets for meeting recordings that was transcribed with accurate speaker labels so that speaker diarization algorithms can be benchmarked.

- **AMI Meeting Corpus** contains around 100 hours of meeting recordings [69]. The corpus contains scenario based as well as real meetings, recorded across three meeting rooms. The raw data was annotated at multiple levels including speech transcriptions, dialog acts and summaries. All meetings are in English with mostly non-native speakers. One or more circular microphone arrays are used to record the data.

- **ICSI Meeting Corpus** consists of 75 meeting recordings that are about one hour in length but range between 17 to 103 minutes [53]. There are 53 unique speakers in the corpus. Each meeting has 3 to 10 participants with an average of 6 per meeting. Meetings are recorded with 4 omni-directional tabletop and 2 electret microphones.

- **ISL Meeting Corpus**: This corpus contains around 100 hours of data across 104 meetings. The set of meetings contain meetings based on different scenarios such as project planning, topic discussion [22]. Initially recordings used single microphone which is increased to three in later recordings.

- **NIST Meeting Pilot Corpus**: This is a small corpus containing 19 meetings between 2001 and '03 with a total of 15 hours [38]. Audio recordings use 3 omni-directional microphones and one circular directional microphone with 4 elements.

Other Meeting corpora include VACE Multimodal corpus [23], CHIL Corpus [77] LDC meeting data [42].

2.5.1 NIST Rich Transcription Evaluations

National Institute for Standards and Technology (NIST) has been organizing a Rich Transcription (RT) evaluation to benchmark the advances in the state-of-the-art in several automatic speech recognition technologies. The evaluation series aims to create technologies that produce transcriptions which are more readable by humans and useful to machines. RT Evaluations has been conducted from 2002 onwards. Speaker diarization evaluations had started and principal focus on

Conversational Telephone Speech and Broadcast News data (2002,'03,'04). Meeting diarization was introduced later and is the prime focus of all evaluations since 2006.

These evaluations use Diarization Error (DER) to measure the performance. It is measured as the fraction of time that is not assigned correctly to a speaker or nonspeech. This measure is computed as follows. Since the sets of speaker labels provided by reference and hypothesis from the system are arbitrary, an optimal mapping between the two sets is first constructed such that the overlap time between corresponding speakers is maximum. The DER is then calculated as:

$$DER = \frac{\sum_{all\ seg}\{dur(seg)[\max(N_{ref}(seg), N_{hyp}(seg)) - N_{correct}(seg)]\}}{\sum_{allseg} dur(seg)N_{ref}(seg)} \tag{2.12}$$

where $N_{ref}(seg)$ and $N_{hyp}(seg)$ denote the number of speakers as determined by the reference and hypothesized segmentation for the speech segment seg with duration $dur(seg)$. $N_{correct}$ denotes the number of speakers that are correctly mapped between reference and hypothesized segmentations. Segments that are labelled as non speech is considered to be having 0 speakers. When all speakers are correctly matched between the reference and hypothesized segmentation for a segment, the corresponding DER is zero.

DER consists of different set of components as follows:

- **False Alarm Speech** is the percentage of scored time that a hypothesized speaker is labelled as non-speech in the reference. It can be expressed as:

$$E_{FA} = \frac{\sum_{seg:N_{hyp}>N_{ref}}\{dur(seg)[N_{hyp}(seg) - N_{ref}(seg)]\}}{\sum_{allseg} dur(seg)N_{ref}(seg)} \tag{2.13}$$

- **Missed Speech** is the percentage of time that a reference speaker segment is mapped to no hypothesized speakers. Missed speech include error thus includes undetected speakers. It is given by:

$$E_{MISS} = \frac{\sum_{seg:N_{hyp}<N_{ref}}\{dur(seg)[N_{hyp}(seg) - N_{ref}(seg)]\}}{\sum_{allseg} dur(seg)N_{ref}(seg)} \tag{2.14}$$

- **Speaker error** represents the percentage of time when a hypothesized speaker label is mapped to the wrong reference speaker label. Speaker error is given by:

$$E_{SPKR} = \frac{\sum_{all\ seg}\{dur(seg)[\min(N_{ref}(seg), N_{hyp}(seg)) - N_{correct}(seg)]\}}{\sum_{allseg} dur(seg)N_{ref}(seg)} \tag{2.15}$$

Diarization error can be re-written as:

$$DER = E_{FA} + E_{MISS} + E_{SPKR} \tag{2.16}$$

The segment boundaries are evaluated upto an accuracy of a predefined collar that accounts for the inexact boundaries in the labelling. This value was fixed by NIST at 0.25s.

DER for a dataset with multiple meetings is computed from individual meeting DER values by performing a linear combination of individual DER values. The weight for each meeting is proportional to the total scored time ($\sum_{allseg} dur(seg)N_{ref}(seg)$) of the meeting.

2.6 Context of work in thesis

As we can see from the review, agglomerative clustering is followed in most of the diarization systems that is based on a similarity measure between segments. Several similarity measures have been considered in the literature as described in Section 2.4.1. The choice of this distance measure is somewhat arbitrary. In this work, we instead try to minimize an objective function that maximizes the mutual information with respect to a set of relevance variables. The distance measure arises from the objective function and depends on the distribution of the relevance variables. This distance measure is used to perform clustering and realignment. The distribution is estimated from different feature streams when multiple feature streams are present.

Chapter 3

Speaker Diarization based on IB Principle

In this chapter we investigate the use of a clustering technique motivated from an information theoretic framework known as the *Information Bottleneck* (IB) [117]. The IB method has been applied to clustering of different types of data like documents [109; 110] and images [43]. IB clustering [106; 117] is a distributional clustering inspired from Rate-Distortion theory [27]. In contrast to many other clustering techniques, it is based on preserving the relevant information specific to a given problem instead of arbitrarily assuming a distance function between elements. Furthermore, given a data set to be clustered, IB tries to find the trade-off between the most compact representation and the most informative representation of the data.

In the rest of the chapter we review Information Bottleneck Principle and different algorithms to optimize IB objective function (Section 3.1). The model selection algorithms are introduced in Section 3.2. The full diarization system is then described (Section 3.4). The experiments and benchmark tests are presented in Sections 3.5 and 3.6.

3.1 Information Bottleneck Principle

The Information Bottleneck (IB) [106; 117] is a distributional clustering framework based on information theoretic principles. It is inspired from the Rate-Distortion theory [27] in which a set of elements X is organized into a set of clusters C minimizing the distortion between X and C. Unlike the Rate-Distortion theory, the IB principle does not make any assumption about the distance between elements of X. On the other hand, it introduces the use of a set of *relevance variables*, Y, which provides meaningful information about the problem. For instance, in a document clustering problem, the relevance variables could be represented by the vocabulary of words. Similarly, in a speech recognition problem, the relevance variables could be represented as the target sounds. IB tries to find the clustering representation C that conveys as much information as possible about Y. In this way the IB clustering attempts to keep the meaningful information with respect to a given problem.

Let Y be the set of variables of interest associated with X such that $\forall x \in X$ and $\forall y \in Y$ the conditional distribution $p(y|x)$ is available. Let clusters C be a compressed representation of input data X. Thus, the information that X contains about Y is passed through the compressed representation (bottleneck) C. The Information Bottleneck (IB) principle states that this clustering representation should preserve as much information as possible about the relevance variables Y (i.e., maximize $I(Y,C)$) under a constraint on the mutual information between X and C i.e. $I(C,X)$. Dually, the clustering C should minimize the coding length (or the compression) of X using C i.e. $I(C,X)$ under the constraint of preserving the mutual information $I(Y,C)$. In other words, IB tries to find a trade-off between the most compact and most informative representation w.r.t. variables Y. This corresponds to maximization of the following criterion:

$$\mathcal{F} = I(Y,C) - \frac{1}{\beta} I(C,X) \tag{3.1}$$

where β (notation consistent with [117]) is the Lagrange multiplier representing the trade off between amount of information preserved $I(Y,C)$ and the compression of the initial representation $I(C,X)$.

Let us develop mathematical expressions for $I(C,X)$ and $I(Y,C)$. The compression of the representation C is characterized by the mutual information $I(C,X)$:

$$I(C,X) = \sum_{x \in X, c \in C} p(x)p(c|x)log\frac{p(c|x)}{p(c)} \tag{3.2}$$

The amount of information preserved about Y in the representation is given by $I(Y, C)$:

$$I(Y,C) = \sum_{y \in Y, c \in C} p(c)p(y|c)log\frac{p(y|c)}{p(y)} \tag{3.3}$$

The objective function $\mathcal{F}$ must be optimized w.r.t the stochastic mapping $p(C|X)$ that maps each element of the dataset X into the new cluster representation C.

This minimization yields the following set of self -consistent equations that defines the conditional distributions required to compute mutual informations (3.2) and (3.3) see [117] for details:

$$\begin{cases} p(c|x) &= \frac{p(c)}{Z(\beta,x)}\exp(-\beta \cdot KL[p(y|x)||p(y|c)]) \\ p(y|c) &= \sum_x p(y|x)p(c|x)\frac{p(x)}{p(c)} \\ p(c) &= \sum_x p(c|x)p(x) \end{cases} \tag{3.4}$$

where $Z(\beta, x)$ is a normalization function and $KL[.,.]$ represents the Kullback-Liebler divergence given by:

$$KL[p(y|x)||p(y|c)] = \sum_{y \in Y} p(y|x)\log\frac{p(y|x)}{p(y|c)} \tag{3.5}$$

We can see from the system of equations (3.4) that as $\beta \rightarrow \infty$ the stochastic mapping $p(c|x)$ becomes a hard partition of X, i.e. $p(c|x)$ can take values 0 and 1 only.

Various methods to construct solutions of the IB objective function include iterative optimization, deterministic annealing, agglomerative and sequential clustering (for exhaustive review see [106]). Here we focus only on two techniques referred to as agglomerative and sequential information bottleneck, which will be briefly presented in the next sections.

3.1.1 Agglomerative Information Bottleneck

Agglomerative Information Bottleneck (aIB) [109] is a greedy approach to maximize the objective function (3.1). The aIB algorithm creates hard partitions of the data. The algorithm is initialized with the trivial clustering of $|X|$ clusters i.e, each data point is considered as a cluster. Subsequently, elements are iteratively merged such that the decrease in the objective function (3.1) at each step is minimum.

The decrease in the objective function $\Delta\mathcal{F}$ obtained by merging clusters c_i and c_j is given by:

$$\Delta\mathcal{F}(c_i, c_j) = (p(c_i) + p(c_j)) \cdot \bar{d}_{ij} \tag{3.6}$$

where $\bar{d}_{ij}$ is given as a combination of two Jensen-Shannon divergences:

$$\bar{d}_{ij} = JS[p(y|c_i), p(y|c_j)] - \frac{1}{\beta} JS[p(x|c_i), p(x|c_j)] \tag{3.7}$$

where JS denotes the Jensen-Shannon (JS) divergence between two distributions and is defined as:

$$JS(p(y|c_i), p(y|c_j)) = \pi_i \, KL[p(y|c_i)||q_Y(y)] + \pi_j \, KL[p(y|c_j)||q_Y(y)] \tag{3.8}$$

$$JS(p(x|c_i), p(x|c_j)) = \pi_i \, KL[p(x|c_i)||q_X(x)] + \pi_j \, KL[p(x|c_j)||q_X(x)] \tag{3.9}$$

with:

$$q_Y(y) = \pi_i \, p(y|c_i) + \pi_j \, p(y|c_j) \tag{3.10}$$

$$q_X(x) = \pi_i \, p(x|c_i) + \pi_j \, p(x|c_j) \tag{3.11}$$

$$\pi_i = p(c_i)/(p(c_i) + p(c_j)) \tag{3.12}$$

$$\pi_j = p(c_j)/(p(c_i) + p(c_j)) \tag{3.13}$$

The objective function (3.1) decreases monotonically with the number of clusters. The algorithm merges cluster pairs until the desired number of clusters is attained. The new cluster c_r obtained by merging the individual clusters c_i and c_j is characterized by:

$$p(c_r) = p(c_i) + p(c_j) \tag{3.14}$$

$$p(y|c_r) = \frac{p(y|c_i)p(c_i) + p(y|c_j)p(c_j)}{p(c_r)} \tag{3.15}$$

It is interesting to notice that the JS divergence is not an arbitrarily introduced similarity measure between elements but a measure that naturally arises from the maximization of the objective function. For completeness we report the full procedure described in [106] in Figure 3.1.1.

At each agglomeration step, the algorithm takes the merge decision based only on a local criterion. Thus aIB is a greedy algorithm and produces only an approximation to the optimal solution

```
Input:
    Joint Distribution $p(x, y)$
    Trade-off parameter $\beta$
Output:
    $C_m$: $m$-partition of $X$, $1 \leq m \leq |X|$
Initialization:
    $C \equiv X$
    For $i = 1 \ldots N$
        $c_i = \{x_i\}$
        $p(c_i) = p(x_i)$
        $p(y|c_i) = p(y|x_i) \forall y \in Y$
        $p(c_i|x_j) = 1$ if $j = i, 0$ otherwise
    For $i, j = 1 \ldots N, i < j$
    Find $\Delta\mathcal{F}(c_i, c_j)$
Main Loop:
    While $|C| > 1$
        $\{i, j\} = \arg\min_{i', j'} \Delta\mathcal{F}(c_i, c_j)$
        Merge $\{c_i, c_j\} \Rightarrow c_r$ in $C$
            $p(c_r) = p(c_i) + p(c_j)$
            $p(y|c_r) = \frac{[p(y|c_i)p(c_i) + p(y|c_j)p(c_j)]}{p(c_r)}$
            $p(c_r|x) = 1, \forall x \in c_i, c_j$
        Calculate $\Delta\mathcal{F}(c_r, c)$, $\forall c \in C$
```

Figure 3.1. Agglomerative IB algorithm (106)

which may not be the global solution to the objective function.

3.1.2 Sequential Information Bottleneck

Sequential Information Bottleneck (sIB) [110] tries to improve the objective function (3.1) in a given partition. Unlike agglomerative clustering, it works with a fixed number of clusters M. The algorithm starts with an initial partition of the space into M clusters $\{c_1, ..., c_M\}$. Then some element x is drawn out of its cluster c_{old} and represents a new singleton cluster. x is then merged into the cluster c_{new} such that $c_{new} = \arg\min_{c \in C} \Delta\mathcal{F}(x, c)$ where $\Delta\mathcal{F}(.,.)$ is as defined in (3.6). It can be verified that if $c_{new} \neq c_{old}$ then $\mathcal{F}(C_{new}) < \mathcal{F}(C_{old})$ i.e., at each step the objective function (3.1) either improves or stays unchanged. This is performed for each $x \in X$. This process is repeated several times until there is no change in the clustering assignment for any input element. To avoid local maxima, this procedure can be repeated with several random initializations. The sIB algorithm is summarized for completeness in Fig 2.

3.2 Model Selection

In typical diarization tasks, the number of speakers in a given audio stream is not a priori known and must be estimated from data. This means that the diarization system has to solve simultaneously two problems: finding the actual number of speakers and clustering together speech from the same speaker. This problem is often cast into a model selection problem. The number of speakers determines the complexity of the model in terms of number of parameters. The model selection criterion chooses the model with the right complexity and thus the number of speakers. Let us consider the theoretical foundation of model selection.

Consider a dataset X, and a set of parametric models $\{m_1, \cdots, m_M\}$ where m_j is a parametric model with n_j parameters trained on the data X. Model selection aims at finding the model $\hat{m}$ such that:

$$\hat{m} = \arg\max_j \{p(m_j|X)\} = \arg\max_j \left[\frac{p(X|m_j)p(m_j)}{p(X)}\right] \tag{3.16}$$

Given that $p(X)$ is constant and assuming uniform prior probabilities $p(m_j)$ on models m_j, maximization of (3.16) only depends on $p(X|m_j)$. In case of parametric modeling with parameter set θ_j,

```
Input:
    Joint Distribution $p(x,y)$
    Trade-off parameter $\beta$
    Cardinality value $M$
Output:
    Partition $C$ of $X$ into $M$ clusters
Initialization:
    $C \leftarrow$ random partition of $X$ into $M$ clusters
    For every $c_i \in C$:
        $p(c_i) = \sum_{x_j \in c_i} p(x_j)$
        $p(y|c_i) = \frac{1}{p(c_i)} \sum_{x_j \in c_i} p(y|x_j)p(x_j)$
        $p(c_i|x_j) = 1$ if $x_j \in c_i$, $0$ otherwise
Main Loop:
    While not Done
        Done $\leftarrow$ TRUE
        For every $x \in X$:
            $c_{old} \leftarrow$ cluster $x$ belongs
            Split $c_{old} \Rightarrow \{c'_{old}, \{x\}\}$
                $p(c'_{old}) = p(c_{old}) - p(x)$
                $p(y|c'_{old}) = \frac{p(y|c_{old})p(c_{old}) - p(y,x)}{p(c'_{old})}$
                $p(c'_{old}|x_i) = 1; \forall x_i \in c_{old}, x_i \neq x$
            $c_{new} = \arg\min_{c \in C} \Delta\mathcal{F}(\{x\}, c)$
            Merge $\{c_{new}, \{x\}\} \Rightarrow c'_{new}$
                $p(c'_{new}) = p(c_{new}) + p(x)$
                $p(y|c'_{new}) = \frac{p(y|c_{new})p(c_{new}) + p(y,x)}{p(c'_{new})}$
                $p(c'_{new}|x_i) = 1; \forall x_i \in c_{new}, x_i = x$
            If $c_{new} \neq c_{old}$
                Done $\leftarrow$ FALSE
```

Figure 3.2. Sequential IB algorithm (106)

e.g. HMM/GMM, it is possible to write:

$$p(X|m_j) = \int p(X, \theta_j|m_j)d\theta_j \tag{3.17}$$

This integral cannot be computed in closed form in the case of complex parametric models with hidden variables (e.g. HMM/GMM). However several approximations for (3.17) are possible, the most popular one being the *Bayesian Information Criterion (BIC)* [101]:

$$BIC(m_j) = log(p(X|\hat{\theta_j}, m_j)) - \frac{p_j}{2} log\, N \tag{3.18}$$

where p_j is the number of free parameters in the model m_j, $\hat{\theta_j}$ is the MAP estimate of the model computed from data X, and N is the number of data samples. The rationale behind (3.18) is straightforward: models with larger numbers of parameters will produce higher values of $log(p(X|\theta_j, m_j))$ but will be more penalized by the term $\frac{p_j}{2} log\, N$. Thus the optimal model is the one that achieves the best trade-off between data explanation and complexity in terms of number of parameters. However, BIC is exact only in the asymptotic limit $N \to \infty$. It has been shown [24] that in the finite sample case, like in speaker clustering problems, the penalty term must be tuned according to a heuristic threshold. In [3; 4; 2], a modified BIC criterion that needs no heuristic tuning has been proposed and will be discussed in more details in Section 3.6.1.

In the case of IB clustering, there is no parametric model that represents the data and model selection criteria based on a Bayesian framework like BIC cannot be applied. Several alternative solutions have been considered in the literature.

Because of the information theoretic basis, it is straightforward to apply the *Minimum Description Length* (MDL) principle [102]. The MDL principle is a formulation of the model selection problem from an information theory perspective. It states that the optimal model m is the one that minimizes the length of the description of the model m and the description length of the data X given the model m [16]. In mathematical formalism, the MDL criterion for model m is given by The optimal model minimizes the following criterion.

$$\mathcal{F}_{MDL}(m) = L(m) + L(X|m) \tag{3.19}$$

A version of MDL criterion for IB clustering is given by (See Appendix A.1 for details) :

$$\mathcal{F}_{MDL} = N[H(Y) - I(Y,C) + H(C)] + N \log \frac{N}{M} \tag{3.20}$$

Similar to the BIC criterion, $N \log \frac{N}{M}$ acts like a penalty term that penalizes codes that uses too many clusters.

When aIB clustering is applied, expression (A.4) is evaluated for each stage of the agglomeration that produces $|X|$ different clustering solutions ranging from each input element considered as a singleton cluster ($|C| = |X|$) to all input elements assigned to one cluster($|C| = 1$). Then the number of clusters that minimizes (A.4) is selected as the actual number of speakers.

Another way of inferring the right number of clusters can be based on the *Normalized Mutual Information (NMI)* $\frac{I(Y,C)}{I(X,Y)}$. The Normalized Mutual Information $\frac{I(Y,C)}{I(X,Y)}$ represents the fraction of original mutual information that is captured by the current clustering representation. This quantity decreases monotonically with the number of clusters (see Figure 3.3). It can also be expected that this quantity will decrease more when dissimilar clusters are merged. Hence, we investigate a simple thresholding of $\frac{I(Y,C)}{I(X,Y)}$ as a possible choice to determine the number of clusters. The threshold is heuristically determined on a separate development data set.

3.3 Applications

Being a general unsupervised learning techniques, Information Bottleneck framework has many applications. Applying IB to any domain requires the knowledge about the relevance variables in that domain. In this section, the applications to different domains are described.

The original application of the principle was in the domain of Document clustering [107]. The algorithm use a a set of word clusters as the relevance variables. The word clusters themselves is learned as part of the algorithm where the roles of X and Y are interchanged in the IB optimization. i.e., Initially a set of word clusters are learned that captures maximum mutual information about the set of documents. The resulting set of word clusters are then used for document clustering. The system uses aIB clustering to perform the optimization. It was also verified that the word clusters achieve good performance in document classification [108]. Significant improvements in

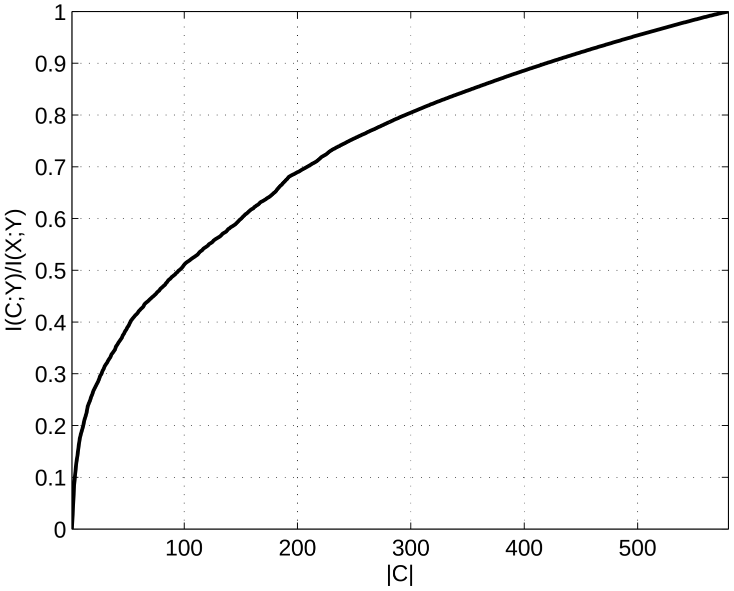


Figure 3.3. Normalized mutual information decreases monotonically with the number of clusters.

classification accuracy was observed even with small training data as compared to just using word distributions for clustering. However aIB algorithm does not scale very well to very large databases due to computational requirements. A sequential optimization (sIB) using the word-clusters as relevance variables is employed in the context of large datasets [111].

A similar double clustering algorithm is used for analyzing neural codes. Neural stimulus features are clustered such that the mutual information of the cluster representation is maximum with respect to the resulting spike trains (relevance variables) [100].

Other applications of IB principle was applied to the image clustering task as well. In case of Image clustering, each image is represented as a set of feature points where the location of the pixel is appended to the color information [43; 44]. This represents the relevance variables. The distribution of relevance variables for each image is represented with a GMM. Since no closed form approximation of KL divergence exists for GMM distributions a Monte-Carlo approximation is used for agglomerative clustering. An alternate representation that avoids the Monte carlo step represents the image with histograms [44]. Discrete distributions are used to perform the IB clustering where a closed form expression exists for the KL divergence.

In the speech processing domain, Information Bottleneck was used for extraction of relevant speech features [47]. The study analyzes Mutual information of the input speech data with respect to two different relevance variable set– phoneme labels and speaker labels. The study reports an increase in the mutual information with respect to phoneme labels with MFCC feature extraction. The study proposes to use task specific features, that maximizes the mutual information with respect to the relevance variables.

3.4 Applying IB to Diarization

To apply the Information Bottleneck principle to the diarization problem, we need to define input variables X to be clustered and the relevance variables Y representing the meaningful information about the input.

In the original case of document clustering, documents represent the input variable X. The vocabulary of words is selected as the relevance variable. Associated conditional distributions $\{p(y_i|x_j)\}$ are the probability of each word y_i in document x_j. Documents can be clustered together with IB using the fact that similar documents will have similar probabilities of containing the same words.

In this paper, we investigate the use of IB for clustering of speech segments according to cluster similarity. We define in the following the input variables $X = \{x_j\}$, the relevance variables $Y = \{y_i\}$ and the conditional probabilities $p(y_i|x_j)$.

3.4.1 Input Variables X

The Short Time Fourier Transform (STFT) of the input audio signal is computed using 30ms windows shifted by a step of 10ms. 19 Mel Frequency Cepstral Coefficients (MFCC) are extracted from each windowed frame. Let $\{s_1, s_2, \cdots s_T\}$ be the extracted MFCC features. Subsequently, a uniform linear segmentation is performed on the feature sequence to obtain segments of a fixed length D (typically 2.5 seconds). The input variables X are defined as the set of these segments $\{x_1, x_2, \cdots, x_M\}$. Thus each segment x_j consists of a sequence of MFCC features $\{s^j_k\}_{k=1,\cdots,D}$.

If the length of the segment is small enough, X may be considered as generated by a single speaker. This hypothesis is generally true in case of Broadcast News audio data. However in case of conversational speech with fast speaker change rate and overlapping speech (like in meeting data), initial segments may contain speech from several speakers.

3.4.2 Relevance Variables Y

Motivated by the fact that GMMs are widely used in speaker recognition and verification systems (see e.g. [94]), we choose the relevant variables $Y = \{y_j\}$ as components of a GMM estimated from the meeting data. A shared covariance matrix GMM is estimated from the entire audio file. The number of components of the GMM is fixed proportional to the length of the meeting i.e. the GMM has $\frac{P}{D}$ components where P is the length of the audio stream (in seconds) and D is length of segments (in seconds) defined in section 3.4.1.

The computation of conditional probabilities $p(Y = y_i|X = x_j)$ is straightforward. Consider a Gaussian Mixture Model $f(s) = \sum_{j=1}^{L} w_j \mathcal{N}(s, \mu_j, \Sigma_j)$ where L is the number of components, w_j are weights, μ_j means and Σ_j covariance matrices. It is possible to project each speech frame s_k onto the space of Gaussian components of the GMM. Adopting the notation used in previous sections, the space induced by GMM components would represent the relevance variable Y.

Computation of $p(y_i|s_k)$ is then simply given by:

$$p(y_i|s_k) = \frac{w_i \mathcal{N}(s_k, \mu_i, \Sigma_i)}{\sum_{j=1}^{L} w_j \mathcal{N}(s_k, \mu_j, \Sigma_j)};\ i = 1, \ldots, L \tag{3.21}$$

The probability $p(y_i|s_k)$ estimates the relevance that the i^{th} component in the GMM has for speech frame s_k. Since segment x_j is composed of several speech frames $\{s_k^j\}$, distributions $\{p(y_i|s_k^j)\}$ can be averaged over the length of the segment to get the conditional distribution $p(Y|X)$.

In other words, a speech segment X is projected into the space of relevance variables Y estimating a set of conditional probabilities $p(Y|X)$.

3.4.3 Clustering

Given the variables X and Y, the conditional probabilities $p(Y|X)$, and trade-off parameter β, Information Bottleneck clustering can be performed. The diarization system involves two tasks: finding the number of clusters (i.e. speakers) and an assignment for each speech segment to a given cluster.

The procedure we use is based on the agglomerative IB described in Section 3.1.1. The algorithm is initialized with M clusters with $M = |X|$ and agglomerative clustering is performed, generating a set of possible solutions in between M and 1 clusters.

Out of the $M = |X|$ possible clustering solutions of aIB, we choose one according to the model selection criteria described in Section 3.2 i.e. *Minimum Description Length* or *Normalized Mutual Information*.

However, agglomerative clustering does not seek the global optimum of the objective function and can converge to local minima. For this reason, the sIB algorithm described in Section 3.1.2 can be applied to improve the partition. Given that sIB works only on fixed cardinality clustering, we propose to use it to improve the greedy solution obtained with the aIB.

To summarize, we study the following four different types of clustering/model selection algorithms:

1 agglomerative IB + MDL model selection.

2 agglomerative IB + NMI model selection.

3 agglomerative IB + MDL model selection + sequential IB.

4 agglomerative IB + NMI model selection + sequential IB.

3.4.4 Diarization algorithm

We can summarize the complete diarization algorithm as follows:

1 Extract acoustic features $\{s_1, s_2, \cdots, s_T\}$ from the audio file.

2 Speech/non-speech segmentation and reject non-speech frames.

3 Uniform segmentation of speech in chunks of fixed size D, i.e. definition of set $X = \{x_1, x_2, \cdots, x_M\}$.

4 Estimation of GMM with shared diagonal covariance matrix i.e. definition of set Y.

5 Estimation of conditional probability $p(Y|X)$.

6 Clustering based on one of the methods described in Section 3.4.3.

7 Viterbi realignment using conventional GMM system estimated from previous segmentation.

Steps 1 and 2 are common to all diarization systems. Speech is segmented into fixed length segments in step 3. This step tries to obtain speech segments that contain speech from only one speaker. We use a uniform segmentation in this work though other solutions like speaker change detection or K-means algorithm could be employed.

Step 4 trains a background GMM model with shared covariance matrix from the entire audio stream. Though we use data from the same meeting, it is possible to train the GMM on a large independent dataset i.e. a Universal Background Model (UBM) can be used.

Step 5 involves conditional probability $p(y|x)$ estimation. In step 6 clustering and model selection are performed on the basis of the Information Bottleneck principle.

Step 7 refines initial uniform segmentation by performing a set of Viterbi realignments. This step modifies the speaker boundaries and is discussed in the following section.

3.4.5 Viterbi Realignment

As described in 3.4.1, the algorithm clusters speech segments of a fixed length D. Hence, the cluster boundaries obtained from the IB are aligned with the endpoints of these segments. Those endpoints are clearly arbitrary and can be improved by re-aligning the whole meeting using a Viterbi algorithm.

The Viterbi realignment is performed using an ergodic HMM. Each state of the HMM represents a speaker cluster. The state emission probabilities are modeled with Gaussian Mixture Models, with a minimum duration constraint. Each GMM is initialized with a fixed number of components.

The IB clustering algorithm infers the number of clusters and the assignment from X segments to C clusters. A separate GMM for each cluster is trained using data assignment produced by the IB clustering. The whole meeting data is then re-aligned using the ergodic HMM/GMM models. During the re-alignment a minimum duration constraint of 2.5 seconds is used as well.

3.5 Effect of System Parameters

In this section we study the impact of the trade-off parameter β (Section3.5.2), the performance of the agglomerative and sequential clustering (Section 3.5.3), the model selection criterion (Section 3.5.4) and the effect of the Viterbi re-alignment (Section 3.5.5) on development data.

3.5.1 Data description

The data used for the experiments consist of meeting recordings obtained using an array of far-field microphones also referred as Multiple Distant Microphones (MDM). Those data contain mainly conversational speech with high speaker change rate and represent a very challenging data set.

We study the impact of different system parameters on the development dataset which contains meetings from previous years' NIST evaluations for "Meeting Recognition Diarization" task [85]. This development dataset contains 12 meeting recordings each one around 10 minutes. The best set of parameters is then used for benchmarking the proposed system against a state-of-the-art diarization system. Comparison is performed on the NIST RT06 evaluation data for "Meeting Recognition Diarization" task . The dataset contains nine meeting recordings of approximately 30 minutes each.

Preprocessing consists of the following steps: signals recorded with Multiple Distant Microphones are filtered using a Wiener filter denoising for individual channels followed by a delay-and-sum beamforming [12],[5]. This was performed using the *BeamformIt* toolkit [147]. Such preprocessing produces a single enhanced audio signal from individual far-field microphone channels. 19 MFCC features are then extracted from the beam-formed signal.

The system performance is evaluated in terms of Diarization Error Rates (DER). DER is the sum of missed speech errors (speech classified as non-speech), false alarm speech error (non-speech classified as speech) and speaker error [84]. Speech/non-speech (spnsp) error is the sum of missed speech and false alarm speech. For all experiments reported in this paper, we include the overlapped speech in the evaluation.

Speech/non-speech segmentation is obtained using a forced alignment of the reference transcripts on close talking microphone data using the AMI RT06 first pass ASR models [45]. Results are scored against manual references force aligned by an ASR system. Being interested in comparing the clustering algorithms, the same speech/non-speech segmentation will be used across all experiments. The missed speech, false alarm speech and total speech/non-speech error for all meetings in the development dataset and evaluation dataset are listed in Table 3.1 and Table 3.2 respectively.

Table 3.1. Missed Speech, False Alarm and Total Speech/Non-speech Error for the Development Dataset

Meeting	Miss	FA	spnsp
AMI_20041210-1052	0.40	1.20	1.60
AMI_20050204-1206	2.60	2.10	4.70
CMU_20050228-1615	9.40	1.10	0.50
CMU_20050301-1415	3.80	1.60	5.40
ICSI_20000807-1000	4.70	0.30	5.00
ICSI_20010208-1430	3.70	1.00	4.70
LDC_20011116-1400	2.10	1.70	3.80
LDC_20011116-1500	5.90	1.00	6.90
NIST_20030623-1409	1.00	0.60	1.60
NIST_20030925-1517	7.70	5.70	3.40
VT_20050304-1300	0.60	1.00	1.60
VT_20050318-1430	1.40	6.20	7.60
ALL	3.50	1.80	5.30

Table 3.2. Missed Speech, False Alarm and Total Speech/Non-speech Error for the Evaluation Dataset

Meeting	Miss	FA	spnsp
CMU_20050912-0900	11.60	0.20	11.80
CMU_20050914-0900	10.30	0.00	10.30
EDI_20050216-1051	4.90	0.10	5.00
EDI_20050218-0900	4.30	0.10	4.40
NIST_20051024-0930	7.00	0.20	7.20
NIST_20051102-1323	6.10	0.10	6.20
TNO_20041103-1130	3.80	0.10	3.90
VT_20050623-1400	5.20	0.20	5.40
VT_20051027-1400	3.50	0.30	3.80
ALL	6.50	0.10	6.60

3.5.2 Trade-off β

The parameter β represents the trade-off between the amount of information preserved and the level of compression. To determine its value, we studied the diarization error of the IB algorithm in the development dataset. The performance of the algorithm is studied by varying β on a log-linear scale and applying aIB clustering. The optimal number of clusters is chosen according to an oracle. Thus, the influence of the parameter can be studied independently of model selection methods or thresholds. The Diarization Error Rate (DER) of the development dataset for different values of beta is presented in Fig 3.4. These results do not include Viterbi re-alignment. The value of $\beta = 10$ produce the lowest DER. In order to understand how the optimal value of β changes across different meetings, we report in Table 3.3 optimal β for each meeting, DER for the optimal β and for $\beta = 10$. In eight meetings out of the twelve, the β that produces the lowest DER is equal to 10.

In four meetings the optimal β is different from 10, but only in one (CMU_20050228) the DER is significantly different from the one obtained using $\beta = 10$. To summarize the optimal value of β seems to be consistent across different meetings.

Table 3.3. Optimal value for β for each meeting in the development dataset. DER for the optimal β as well as $\beta = 10$ are reported.

Meeting	optimal β	DER at optimal β	DER at $\beta = 10$
AMI_20041210-1052	10	4.6	4.6
AMI_20050204-1206	10	10.0	10.0
CMU_20050228-1615	50	20.4	25.3
CMU_20050301-1415	10	9.4	9.4
ICSI_20000807-1000	100	11.9	12.3
ICSI_20010208-1430	10	12.9	12.9
LDC_20011116-1400	1000	6.2	8.7
LDC_20011116-1500	10	18.7	18.7
NIST_20030623-1409	10	6.0	6.0
NIST_20030925-1517	10	24.3	24.3
VT_20050304-1300	10	7.3	7.3
VT_20050318-1430	100	28.5	29.7

Figure 3.5 shows the DER curve w.r.t. number of clusters for two meetings (LDC_20011116-1400 and CMU_20050301-1415). It can be seen that the DER is flat for $\beta = 1$ and does not decrease with the increase in number of clusters. This low value of β implies more weighting to the regularization term $\frac{1}{\beta}I(C,X)$ of the objective function in Equation (3.1). Thus the optimization tries to minimize $I(C,X)$. The algorithm uses hard partitions i.e. $p(c|x) \in \{0,1\}$, this leads to $H(C|X) = -\sum_{x\in X} p(x) \sum_{c\in C} p(c|x) \log p(c|x) = 0$ and as a result $I(C,X) = H(C) - H(C|X) = H(C)$. Hence minimizing $I(C,X)$ is equivalent to minimizing $H(C)$. Thus $H(C)$ is minimized while clustering with low values of β. This leads to a highly unbalanced distribution where most of the elements are assigned to one single cluster($H(C) \approx 0$). Thus the algorithm always converges towards one large cluster followed by several spurious clusters and the DER stays almost constant. Conversely, when β is high (eg: $\beta = \infty$), effect of this regularization term vanishes. The optimization criterion focuses only on the relevance variable set $I(Y,C)$ regardless of the data compression. The DER curve thus becomes less smooth.

For intermediate values of β, the clustering seeks the most informative *and* compact representation. For the value of $\beta = 10$, the region of low DER is almost constant for comparatively more

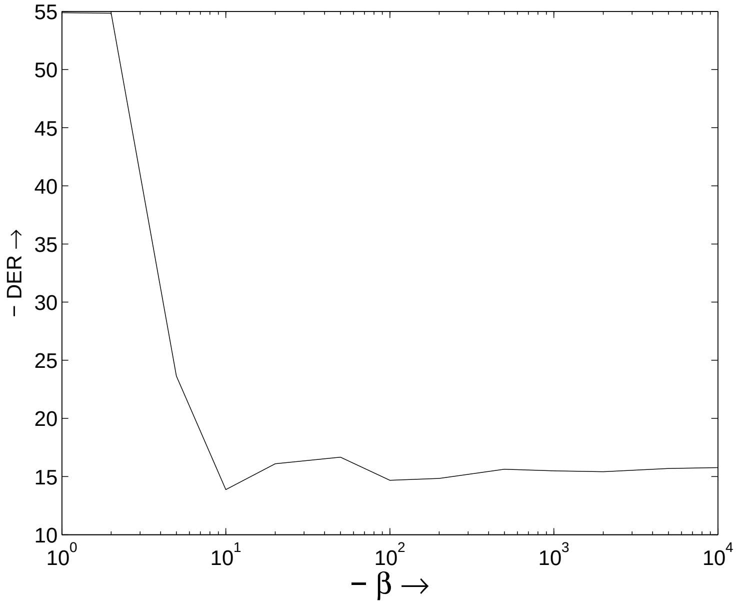


Figure 3.4. Effect of varying parameter β on the diarization error for the development dataset. The optimal β is chosen as $\beta = 10$

values of $|C|$. In this case, the algorithm forms large speaker clusters initially. Most of the remaining clusters are small and merging these clusters does not change the DER considerably. This results in a regularized DER curve as a function of number of clusters (see Figure 3.5).

3.5.3 Agglomerative and Sequential clustering

In this section, we compare the agglomerative and sequential clustering described in Sections 3.1.1, 3.1.2 on the development data. As before model selection is performed using an oracle and the value of β is fixed at 10 as found in the previous section. Agglomerative clustering achieves a DER of 13.3% while sequential clustering achieves a DER of 12.4%, i.e. 1% absolute better. Results are presented in Table 3.4. Improvements are obtained on 8 of the 12 meetings included in the development data.

Also the additional computation introduced by the sequential clustering is small when initialized with aIB output. The sIB algorithm converges faster in this case than using random initial partitions (4 iterations as compared to 6 iterations on an average across the development dataset).

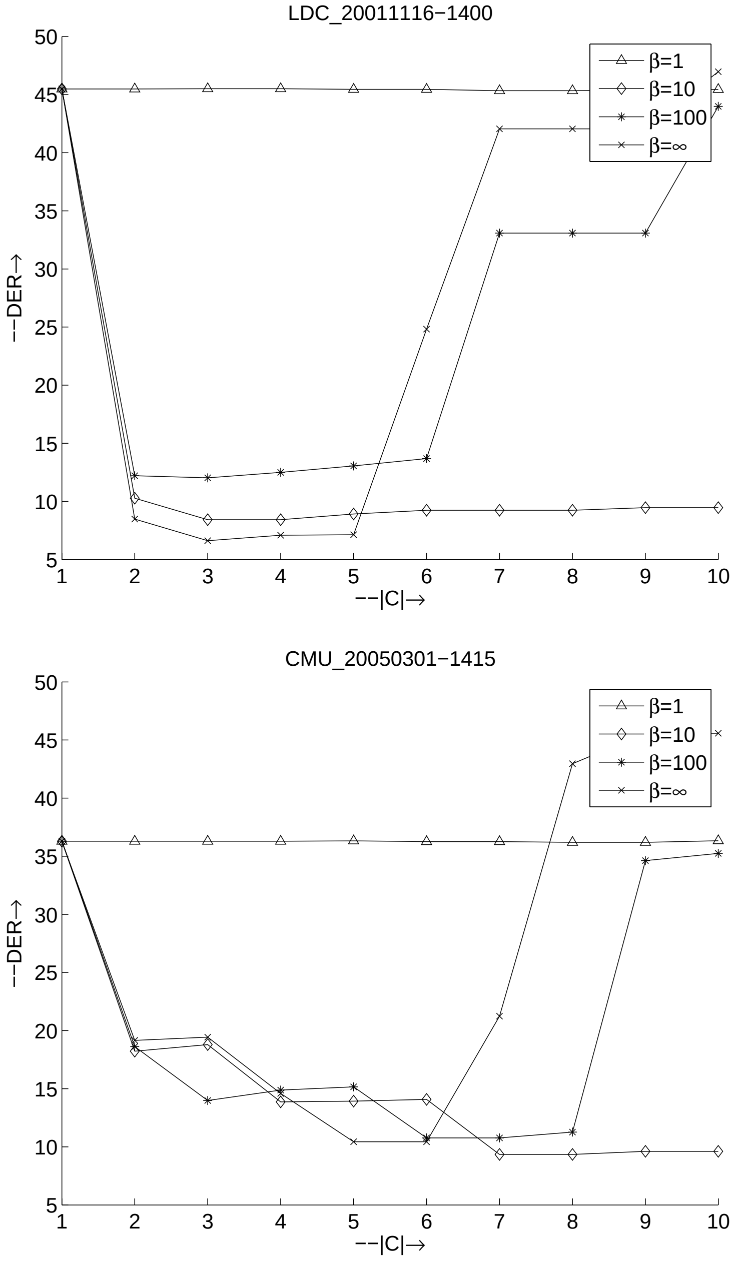


Figure 3.5. DER as a function of number of clusters ($|C|$) for different values of parameter β

Table 3.4. Diarization error rate of development data for individual meetings for aIB and aIB+sIB using oracle model selection and without Viterbi re-alignment.

		aIB
Meeting	aIB	+ sIB
AMI_20041210-1052	4.6	3.7
AMI_20050204-1206	10.0	8.3
CMU_20050228-1615	25.3	25.2
CMU_20050301-1415	9.4	10.1
ICSI_20000807-1000	12.3	13.2
ICSI_20010208-1430	12.9	13.0
LDC_20011116-1400	8.7	7.0
LDC_20011116-1500	18.7	17.5
NIST_20030623-1409	6.0	5.7
NIST_20030925-1517	24.3	23.9
VT_20050304-1300	7.3	5.2
VT_20050318-1430	29.7	25.6
ALL	13.3	12.4

3.5.4 Model selection

In this section, we discuss experimental results with the model selection algorithms presented in Section 3.2. Two different model selection criteria – Normalized Mutual Information (NMI) and Minimum Description Length (MDL) – are investigated to select the number of clusters. They are compared with an oracle model selection which manually chooses the clustering with the lowest DER. The Normalized Mutual Information is a monotonically increasing function with the number of clusters. The NMI value is compared against a threshold to determine the optimal number of

Table 3.5. Optimal value for NMI threshold for each meeting in the development dataset. The DER is reported for the optimal value as well as for 0.3. The clustering is performed with $\beta = 10$

	optimal NMI	DER at	DER at
Meeting	threshold	opt th.	thres 0.3
AMI_20041210-1052	0.3	9.6	9.6
AMI_20050204-1206	0.3	14.9	14.9
CMU_20050228-1615	0.3	26.5	26.5
CMU_20050301-1415	0.3	9.6	9.6
ICSI_20000807-1000	0.4	13.5	20.0
ICSI_20010208-1430	0.3	14.4	14.4
LDC_20011116-1400	0.3	9.2	9.2
LDC_20011116-1500	0.2	20.6	21.9
NIST_20030623-1409	0.4	7.8	11.9
NIST_20030925-1517	0.4	25.2	30.6
VT_20050304-1300	0.3	5.9	5.9
VT_20050318-1430	0.3	34.9	34.9

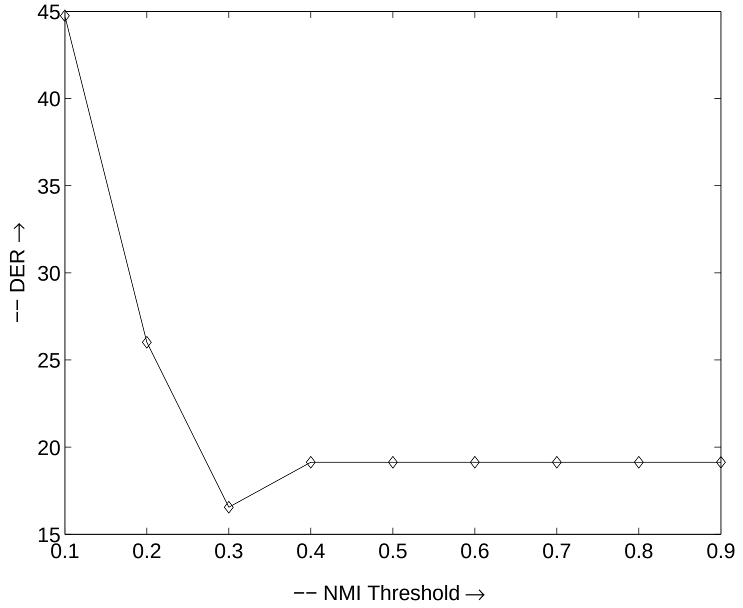


Figure 3.6. Effect of varying NMI threshold on the diarization error for the development dataset. The optimal threshold is fixed as 0.3

clusters in the model. Figure 3.6 illustrates the change of overall DER over the whole development dataset for changing the value of this threshold. The lowest DER is obtained for the value of 0.3. In order to understand how the optimal value of the threshold changes across different meetings, we report in Table 3.5 optimal threshold for each meeting, DER for the optimal threshold and for threshold equal to 0.3. In eight out the twelve meetings in the development data set, the threshold that produces the lowest DER is equal to 0.3. Only in two meetings (ICSI_20000807-1000 and NIST_20030925-1517) results obtained with the optimal threshold are significantly different from those obtained with the value 0.3.To summarize the optimal value of the threshold seems to be consistent across different meetings.

The MDL criterion described in equation (A.4) is also explored for performing model selection. Speaker error rates corresponding to both the methods are reported in Table 3.6. The NMI criterion outperforms the MDL model selection by $\sim 2\%$. The NMI criterion is 2.5% worse than the oracle model selection.

Table 3.6. Diarization Error Rates for dev dataset with NMI, MDL and oracle model selection.

	aIB		aIB+sIB	
Model selection	without Viterbi	with Viterbi	without Viterbi	with Viterbi
Oracle	13.3	10.3	12.4	10.0
MDL	17.3	14.3	16.2	13.8
NMI	15.4	12.6	14.3	12.5

3.5.5 Viterbi realignment

The Viterbi realignment is carried out using an ergodic HMM as discussed in Section 3.4.5. The number of components of each GMM is fixed at 30 based on experiments on the development dataset. The performance after Viterbi realignment is presented in Table 3.6. The DER is reduced by roughly 3% absolute for all the different methods. The lowest DER is obtained using sequential clustering with NMI model selection.

3.6 RT06 Meeting Diarization

In this section we compare the IB system with a state-of-the-art diarization system based on HMM/GMM. Results are provided for the NIST RT06 evaluation data. Section 3.6.1 describes the baseline system while Section 3.6.2 describes the results of the IB based system. Section 3.6.3 compares the computational complexity of the two systems.

3.6.1 Baseline System

The baseline system is an ergodic HMM as described in [3; 5]. Each HMM state represents a cluster (speaker). The state emission probabilities are modeled by Gaussian Mixture Models (GMM) with a minimum duration constrain of 3 seconds. 19 MFCC coefficients extracted from the beam-formed signal are used as the input features. The algorithm follows an agglomerative framework, i.e, it starts with a large number of clusters (hypothesized speakers) and then iteratively merges similar clusters until it reaches the best model. After each merge, data are re-aligned using a Viterbi algorithm to refine speaker boundaries.

The initial HMM model is built using uniform linear segmentation and each cluster is modeled with a 5 component GMM. The algorithm then proceeds with bottom-up agglomerative clustering of the initial cluster models [24]. At each step, all possible cluster merges are compared using a modified version of the BIC criterion[101; 3] which is described below.

Consider a pair of clusters c_i and c_j with associated data D_i and D_j respectively. Also let the number of parameters for modeling each cluster respectively be p_i and p_j parameterized by the GMM models m_i and m_j. Assume the new cluster c having data D obtained by merging D_i and D_j is modeled with a GMM model m parameterized by p Gaussians. The pair of clusters that results in the maximum increase in the BIC criterion (given by equation 3.18) are merged.

$$(i', j') = \arg\max_{i,j} BIC(m) - [BIC(m_j) + BIC(m_i)] \tag{3.22}$$

In [3], the model complexity (i.e. the number of parameters) before and after the merge is made the same. This is achieved by keeping the number of Gaussians in the new model m the same, i.e, as the sum of number of Gaussians in m_j and m_i. i.e., $p = p_i + p_j$. Under this condition equation (3.22) reduces to

$$(i', j') = \arg\max_{i,j} \log \frac{p(D|m)}{p(D_i|m_i)p(D_j|m_j)} \tag{3.23}$$

This eliminates the need of the penalty term from the BIC. Following the merge, all cluster models are updated using an EM algorithm. The merge/re-estimation continues until no merge results in any further increase in the BIC criterion. This determines the number of clusters in the final model. This approach yields state-of-the art results [5] in several diarization evaluations. The performance of the baseline system is presented in Table 5.1. The table lists missed speech, false alarm, speaker error and diarization error. [1]

Table 3.7. Results of the baseline system

File	Miss	FA	spnsp	spkr err	DER
All meetings	6.5	0.1	6.6	17.0	23.6

[1] We found that one channel of the meeting in RT06 denoted with VT_20051027-1400 is considerably degraded. This channel was removed before beamforming. This produces better results for both baseline and IB systems compared to those presented in [131].

3.6.2 Results

In this section we benchmark the IB based diarization system on RT06 data. The same speech/non-speech segmentation is used for all methods. According to the results of previous sections the value of β is fixed at 10. The NMI threshold value is fixed at 0.3. Viterbi re-alignment of the data is performed after the clustering with a minimum duration constrain of $2.5s$ to refine cluster boundaries. Since we use the same speech/non-speech segmentation the values of false alarm and missed speech are same as Table 5.1 and only speaker error values are reported henceforth.

Table 3.8 reports results for aIB and aIB+sIB clustering. Results for both NMI and MDL criteria are reported. NMI is more effective than MDL by 0.7%. Sequential clustering (aIB+sIB) outperforms agglomerative clustering by 0.5%. As in the development data, the best results are obtained by aIB+sIB clustering with NMI model selection. This system achieves a DER of 16.6% as compared to 17.0% for the baseline system.

Table 3.8. Speaker error values for RT06 evaluation data.

Model selection	aIB+ Viterbi	sIB+ Viterbi
MDL	17.8	17.2
NMI	17.1	**16.6**

Table 3.9 reports diarization error for individual meetings of the RT06 evaluation data set. We can observe that overall performances are very close to those of the baseline system but results per meeting are quite different. This difference can be mainly attributed to the different optimization criteria used by the two systems – BIC criterion for the baseline system and IB criterion for the proposed system.

Furthermore, the IB clustering is based on the use of a set of relevance variables defined as the components of a background GMM. The GMM is estimated using data from the same meeting. As variations in signal properties like Signal-to-noise-ratio (SNR) and amount of overlapping speech can deteriorate the quality of the GMM thus the clustering results. For instance, the performance of the IB system are comparatively low for CMU meetings which contain large amounts of overlapping speech and low SNR. On the other hand, IB performs considerably better then the baseline system on VT meetings that have high SNR and TNO meeting which has very less overlapping speech.

Table 3.10 shows the number of speakers estimated by different algorithms for the RT06 eval

Table 3.9. Speaker error for individual meetings using NMI model selection.

		Viterbi realign	
Meeting	Baseline	aIB	aIB + sIB
CMU_20050912-0900	6.0	8.3	6.9
CMU_20050914-0900	5.0	11.6	10.5
EDI_20050216-1051	41.0	43.5	45.5
EDI_20050218-0900	19.4	28.9	28.7
NIST_20051024-0930	4.8	9.0	10.1
NIST_20051102-1323	17.5	9.5	8.8
TNO_20041103-1130	27.6	24.8	22.2
VT_20050623-1400	19.0	4.2	4.0
VT_20051027-1400	17.9	16.2	14.6

Table 3.10. Estimated number of speakers by different model selection criteria.

		aIB + sIB		HMM/
Meeting	#speakers	NMI	MDL	GMM
CMU_20050912-0900	4	5	5	5
CMU_20050914-0900	4	6	6	5
EDI_20050216-1051	4	7	7	5
EDI_20050218-0900	4	7	7	6
NIST_20051024-0930	9	7	7	5
NIST_20051102-1323	8	7	7	6
TNO_20041103-1130	4	7	6	5
VT_20050623-1400	5	8	8	5
VT_20051027-1400	4	6	4	4

data. The number of speakers in the IB system is mostly higher than the actual. This is due to the presence of small spurious clusters with very short duration (typically less than 5 seconds). However those small clusters does not significantly affect the final DER which is similar to the HMM/GMM system.

3.6.3 Computational Complexity

Both the The IB bottleneck algorithm and the baseline HMM/GMM system use the agglomerative clustering framework. Let the number of clusters at a given step in the agglomeration be K. At each step, the agglomeration algorithm needs to calculate the distance measure between each pair of clusters. i.e., $\frac{1}{2}K(K-1)$ distance calculations. Let us consider the difference between the two methods:

- In the HMM/GMM model, each distance calculation involves computing the BIC criterion as given by equation (3.23). Thus a new parametric model m has to be estimated for every possible merge. This requires training a GMM model for every pair of clusters. The training is done using the EM algorithm which is computationally demanding. In other words, this method involves the use of EM parameter estimation for every possible cluster merge.

- In the IB framework, the distance measure is the sum of two Jensen-Shannon divergences as described by equation (3.7). The JS divergence calculation is straightforward and computationally very efficient. Thus the distance calculation in the IB frame work is much faster as compared to the HMM/GMM approach. The distribution obtained merging two clusters is given by equations (3.14-3.15) which simply consists in averaging distributions of individual clusters.

In summary while the HMM/GMM systems make intensive use of the EM algorithm, the IB based system performs the clustering in the space of discrete distributions using closed form equations for distance calculation and cluster distribution update. Thus the proposed approach require less computation than the baseline.

We perform benchmark experiments on a desktop machine with AMD Athlon$^{\text{TM}}$ 2.4GHz 64 X2 Dual Core Processor and 2GB RAM. Table 3.11 lists the real time factors for the baseline and IB based diarization systems for the RT06 meeting diarization task. It can be seen that the IB based systems are significantly faster than HMM/GMM based system. Note that most of the algorithm time for IB systems is consumed for estimating the posterior features. The clustering is very fast and takes only around 30% of the total algorithm time. Also, introducing the sequential clustering contributes very little to the total algorithm time ($\approx 8\%$). Overall the proposed diarization system is considerably faster than-real time.

Table 3.11. Real time factors for different algorithms on RT06 eval data

method	posterior calculation	clustering	Viterbi realign	Total
aIB	0.09	0.06	0.07	0.22
aIB +sIB	0.09	0.08	0.07	0.24
Baseline	–	–	–	3.5

3.7 Discussions

We have presented speaker diarization systems based on the information theoretic framework known as the Information Bottleneck. This system can achieve Diarization Error rates close to those obtained with conventional HMM/GMM agglomerative clustering. In the following we discuss main differences between this framework and traditional approaches.

- *Distance measure*: in the literature, several distance measures have already been proposed for clustering speakers e.g. BIC, generalized log-likelihood ratio, KL divergence and cross-likelihood distances. The IB principle states that when the clustering seeks the solution that preserves as much information as possible w.r.t a set of relevance variables, the optimal distance between clusters is represented by the *Jensen-Shannon* divergence (see equation 3.8). JS divergence can be written as the sum of two KL divergences and has many appealing properties related to Bayesian error (see [66] for detailed discussion). This similarity measure between clusters is not arbitrarily introduced but is naturally derived from the IB objective function (see [109]).

- *Regularization*: The trade-off parameter β between amount of mutual information and compression regularizes the clustering solution as shown in Section 3.5.2. We verified that this term can reduce the DER and make the DER curve more smooth against the number of clusters.

- *Parametric Speaker Model*: HMM/GMM based systems build an explicit parametric model for each cluster and for each possible merge. This assumes that each speaker provides enough data for estimating such a model. On the other hand, the system presented here is based on the distance between clusters in a space of relevance variables without any explicit speaker model. The set of relevance variables is defined through a GMM estimated on the entire audio stream. Furthermore the resulting clustering techniques are significantly faster than conventional systems given that merges are estimated in a space of discrete probabilities.

- *Sequential clustering*: Conventional systems based on agglomerative clustering (aIB) can produce sub-optimal solutions due to their greedy nature. Conversely, sequential clustering (sIB) seeks a global optimum of the objective function. In Sections 3.5.3 and 3.6.2, it is shown that sequential clustering outperforms agglomerative clustering by $\sim 1\%$ on development and $\sim 0.5\%$ evaluation data sets. The sequential clustering can be seen as a "purification" algorithm. In the literature, methods aiming at obtaining clusters that contain speech from a single speaker are referred to as "purification" methods. They refine the agglomerative solution according to smoothed log-likelihood [146] or cross Expectation-Maximization between models [82] for finding frames that were wrongly assigned. In case of sIB, the purification is done according to the same objective function, and the correct assignment of each speech segment is based on the amount of mutual information it conveys on the relevance variables. Furthermore, as reported in Table 3.11, its computational complexity is only marginally higher than the one obtained using agglomerative clustering.

Thus the proposed system based on the IB principle can achieve on RT06 evaluation data a DER of 23.2% as compared to 23.6% of HMM/GMM baseline while running 0.3xRT i.e. significantly faster than the baseline system. In the next chapter we extend this system to handle multiple feature streams.

Chapter 4

Multistream Diarization

Typical acoustic features consist of short term spectral features such as Mel Frequency Cepstral Coefficients (MFCC). In the meeting scenario, data recordings are commonly carried out in a non-intrusive way with multiple distant microphones. The spatial redundancy of the different signals can be used for speaker diarization. For instance, whenever the geometry of the microphone array is known, the speaker locations can be estimated and used as complementary features to conventional MFCC [65]. Otherwise if the array geometry is unknown, the estimated time difference of arrival (TDOA) between different channels of a microphone array can be used as features. Experiments have shown that [89] TDOA performs poorly as stand alone features with respect to MFCC, but significant performance improvements are obtained when MFCC and TDOA are used in combination [88; 90].

MFCC and TDOA are modeled separately with different GMMs and they are combined by linearly weighting the individual log-likelihoods [88]. The log-likelihood combination is used to calculate the BIC distance measure and to refine the speaker boundaries using the Viterbi realignment. The weights of the linear combination are estimated from an independent development data set. This approach has been proven very effective in several evaluations and is implemented in large number of diarization systems [11; 142; 129].

Speaker diarization is applied to recordings performed with varying the number of microphones across different meeting rooms (from 2 microphones to 16 for conference room meetings [32; 33] and up to 64 microphones in case of lecture recordings). If a recording is done with an array of C

microphones, the number of TDOA features is equal to $C - 1$. As a consequence, the dimension of the TDOA feature vector will vary according to the number of microphones resulting in different ranges of log-likelihoods.

In this chapter, we explore the combination of MFCC and TDOA features for speaker diarization in the context of an HMM/GMM system as well as the IB system. The study investigates the issues related with varying dimension of TDOA features on diarization. The chapter include two new contributions to the IB system proposed in Chapter 3.

- The clustering in the space of relevance variables is extended to handle multiple feature streams (Section 4.2). In contrary to the HMM/GMM, it avoids the combination of log-likelihoods.

- The HMM/GMM realignment is replaced with a Kullback-Leibler based realignment as it arises from the IB principle (Section 4.3). The realignment scheme operates on the same relevance variable space and again avoids the combination of log-likelihoods.

The rationale behind performing clustering and realignment using the IB framework rather than log-likelihood combination is that the system should gain robustness to the statistics of the different features (MFCC and TDOA). The proposed approaches are validated in experiments using the same dataset where the number of microphones vary between 2 and 16.

4.1 MFCC and TDOA Combination in HMM/GMM System

Let us consider a diarization system based on the HMM/GMM framework. Each speaker is modeled with an HMM state with minimum duration of three seconds. The system is initialized with an over determined number of speakers by means of uniform segmentation or by speaker change detection methods. Multiple iterations of clustering and realignment are then performed. The clustering follows an agglomerative framework in which a a modified BIC measure [4] is used as the merge/stop criterion.

The BIC criterion depends on the emission probability likelihoods (Equation 3.23) for each cluster. The emission probability distribution b_{c_i} corresponding to speaker cluster c_i is modeled as a

GMM:

$$\log b_{c_i}(s_t) = \log \sum_r w^r_{c_i} \mathcal{N}(s_t, \mu^r_{c_i}, \Sigma^r_{c_i}) \tag{4.1}$$

where s_t is the input feature, $\mathcal{N}(.)$ is the Gaussian pdf and $w^r_{c_i}$, $\mu^r_{c_i}$, $\Sigma^r_{c_i}$ are the weights, means and covariance matrices corresponding to r^{th} mixture Gaussian of cluster c_i.

The pdf of a multidimensional Gaussian distribution with mean $\mu = [\mu_1, \ldots, \mu_d]'$, diagonal covariance matrix $\Sigma = diag(\sigma_1^2, \ldots, \sigma_d^2)$ for an input feature vector $s = [s_1, \ldots, s_d]'$ is given by:

$$\mathcal{N}(s, \mu, \Sigma) = (2\pi)^{-\frac{d}{2}} \left(\Pi_{q=1}^{d} \sigma_q^2\right)^{-\frac{1}{2}} \exp\left(-\Sigma_{q=1}^{d} \frac{(s_q - \mu_q)^2}{2\sigma_q^2}\right) \tag{4.2}$$

where d denotes the feature dimension. If the feature vector has a large dimension, the number of terms (each term is positive) in the summation increases. This would decrease the likelihood (increase the negative log likelihood). Thus the range of log likelihood in Equation 4.1 varies with change in feature dimension.

Each cluster merge is followed by a Viterbi re-alignment that smooths the speaker boundaries and improves the diarization performance. The entire meeting is then realigned with the estimated speaker models after the merge. The oracle path (speaker sequence) $\mathbf{c} = (c_1, \ldots, c_T)$ is determined as the best sequence of states that maximizes the data likelihood. This can be represented as the following optimization:

$$\mathbf{c}^{opt} = \arg\min_{c} \sum_t [-\log b_{c_t}(s_t) - \log(a_{c_t c_{t+1}})] \tag{4.3}$$

where c_t is the speaker cluster at time t. The term $a_{c_i c_j}$ represents the transition probability from speaker state c_i to c_j. The transition probabilities incorporate a minimum duration constraint.

4.1.1 Multiple Feature Streams

Whenever multiple feature streams are available, the HMM/GMM system uses a linear combination of log likelihoods. This approach models the two feature streams with separate GMMs and the combination is then performed by linearly weighting their log-likelihoods [88]. GMMs are estimated separately with observations assigned to the same speaker cluster. The individual weights for different feature streams are estimated minimizing the diarization error on a independent development data set.

Let s_t^{mfcc} and s_t^{tdoa} represent the feature values at time t. GMM models $b_{c_i}^{mfcc}(.)$ and $b_{c_i}^{tdoa}(.)$ are estimated separately from MFCC and TDOA features assigned to the same cluster. A linear combination of the log likelihoods is computed as:

$$\log L_{c_k}(s_t) = P_{mfcc} \log b_{c_i}^{mfcc}(s_t^{mfcc}) + P_{tdoa} \log b_{c_i}^{tdoa}(s_t^{tdoa}) \tag{4.4}$$

P_{mfcc} and P_{tdoa} denote the weights corresponding to MFCC and TDOA features respectively such that $P_{mfcc} + P_{tdoa} = 1$. The diarization system then performs both agglomerative clustering and Viterbi realignment using the combination of the log-likelihoods as in Equation (4.4).

4.1.2 Baseline Experiments and Results

In this section the impact of variations in feature statistics in the baseline system is investigated. The RT09 dataset recorded using Multiple Distance Microphones (MDM) across five different meeting rooms is used. The set of meetings with associated number of microphones is listed in Table 4.1.

A delay and sum beamforming [12] is performed on the MDM data to obtain a single enhanced channel. The beamforming is performed with the *BeamformIt* [147] toolkit. The beamforming first selects a reference channel based on maximum average cross correlation with other channels. Then the Time Delay of Arrival (TDOA) of each channel with respect to the reference channel is computed with a generalized cross correlation phase transform (GCC-PHAT). After choosing a reference channel, signal in each channel is windowed using a $500ms$ window. The GCC-PHAT between two channels $s_i[n]$ and $s_j[n]$is defined as

$$G_{PHAT}(f) = \frac{S_i(f) S_j^*(f)}{|S_i(f)||S_j(f)|} \tag{4.5}$$

where $S_i(f)$ and $S_j(f)$ are the Fourier transforms of the two signals. The TDOA of channels s_i with respect to reference channel s_j is estimated as

$$d_{PHAT}(i,j) = \arg\max_{d} R_{PHAT}(d) \tag{4.6}$$

Table 4.1. List of meeting used for evaluation in the paper with associated number of microphones

sl.no.	meeting id	#microphones
1	CMU_20050912-0900	2
2	CMU_20050914-0900	2
3	EDI_20050216-1051	16
4	EDI_20050218-0900	16
5	NIST_20051024-0930	7
6	NIST_20051102-1323	7
7	TNO_20041103-1130	9
8	VT_20050623-1400	4
9	VT_20051027-1400	3

where $R_{PHAT}(d)$ is the inverse Fourier transform of $G_{PHAT}(f)$. Since delay features are calculated with respect to a reference channel number of delay features is one less than the number of microphones. Following the TDOA estimation, a weighted delay and sum combination of all channels results in a single enhanced channel. 19 MFCC coefficients are estimated from this enhanced output using a $30ms$ window shifted every $10ms$. Both MFCC and TDOA values have the same frame rate.

In order to consider the different statistical properties of the features, MFCC are initially modeled with a five component GMM while TDOA are modeled with a single Gaussian [88]. Figure 4.1 plots the average negative log likelihood values of two independent GMMs trained on MFCC and TDOA features for the 9 meetings used in this work. It can be seen that their dynamic ranges are quite different. TDOA likelihoods depend on the feature vector dimensions and thus on the number of microphones. Larger feature dimension leads to larger likelihood values. For example meeting 3 and 4 have the largest feature dimension (16) among the meetings and posses highest negative log likelihood values. Furthermore TDOA and MFCC statistics are considerably different.

Also there is a two order magnitude difference between the minimum and the maximum values of TDOA log likelihoods across different meetings in Figure 4.1. Possible reasons of such variations include variable dimension of features, differences in the recording environments etc.

Let us now consider the effect of this in a diarization system. Since we use same speech non-speech segmentation that has same speech/non-speech error across all experiments (see Table 3.2) only speaker error rates are reported for the purpose of comparison.

The baseline HMM/GMM system is initialized with 16 clusters obtained with uniform linear segmentation and the clustering is performed using modified BIC as the distance measure [142].

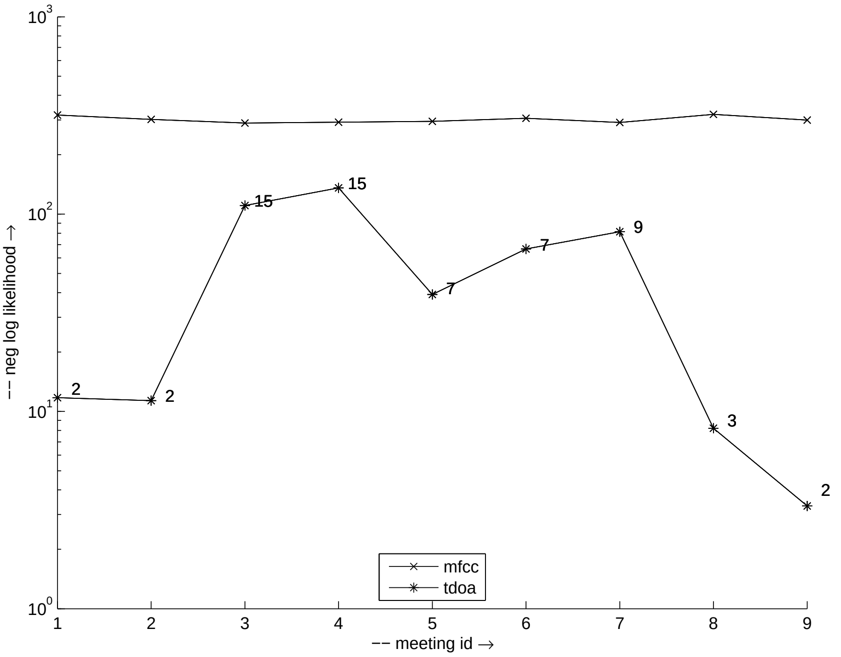


Figure 4.1. Average negative log likelihood values of MFCC and TDOA features at the beginning of clustering for a set of nine meetings. The numbers in the TDOA plot denote the feature dimension.

To study the performances of the diarization systems on the evaluation data, independently of any weight estimation scheme or development data, two different types of oracle experiments referred as meeting-wise oracle and global oracle are proposed.

The meeting-wise oracle is the best possible weighting that yields the minimum speaker error for *each meeting*. It provides the best meeting-wise performance using the feature combination scheme. This is obtained by varying the individual feature weight P_{mfcc} between zero and one such that $P_{mfcc} + P_{tdoa} = 1$ for each meeting m and selecting the set of weight values that minimizes the speaker error. This results in a set of weights $\{P^m_{mfcc}, P^m_{tdoa}\}$ for each meeting recording.

The global oracle is the best possible set of weights *across all meetings* that correspond to the least diarization error in the dataset. In other words, the same set of weights is applied to all the meetings. This is obtained by varying the individual features weight P_{mfcc} between zero and one such that $P_{mfcc} + P_{tdoa} = 1$. The set of values $\{P_{mfcc}, P_{tdoa}\}$ that minimizes the speaker error for the *entire* data set is selected.

The oracle experiments on the evaluation dataset permit the study of the two systems under the ideal weighting that minimizes the speaker error. The comparison between meeting-wise and global

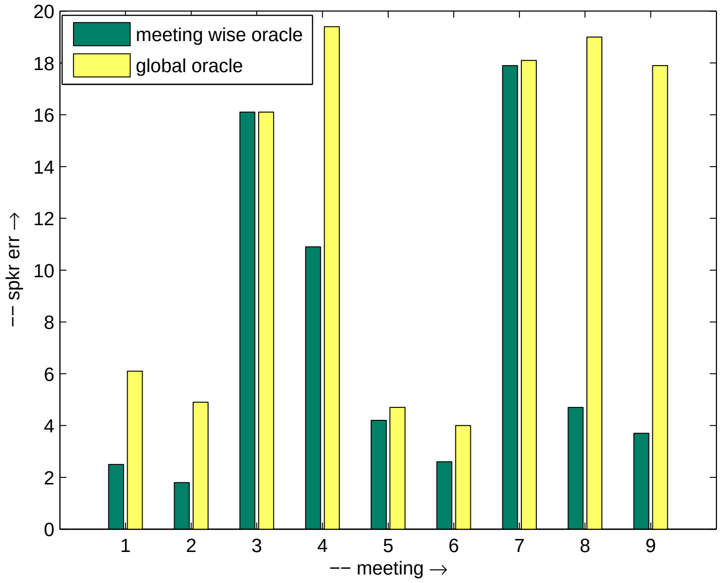


Figure 4.2. Meeting-wise speaker error for MFCC+TDOA feature combination of HMM/GMM system: meeting wise and global oracle weights for each meeting.

Table 4.2. Overall speaker error for MFCC+TDOA combination of the HMM/GMM system: meeting-wise and oracle weights as well as the estimated weight from development data ($P_{tdoa} = 0.1$). $P_{mfcc} = 1 - P_{tdoa}$. The table also reports the speaker error with and without re-alignment after the last clustering step.

meeting-wise oracle	global oracle $P_{tdoa} = 0.01$	Estimated wt. $P_{tdoa} = 0.1$
7.0	11.7	13.6

oracles could provide insights to robustness of the individual feature weights. The corresponding results are compared against the performance of the feature weights estimated from development data ($P_{tdoa} = 0.1$ and $P_{mfcc} = 0.9$ as reported by [88]).

Table 4.2 reports the speaker error for the meeting-wise and global oracle weights, as well as for the estimated weight from development dataset. It can be observed that there is a performance reduction of 4.7% with a global oracle weight as compared to meeting-wise oracle.

The individual meeting performances are depicted in Figure 4.2 and the corresponding oracle weights for the TDOA feature stream are illustrated in Figure 4.3. It can be seen that:

1. the magnitudes of the weights span a considerably large range (note that the plot is in logarithmic scale). This could happen due to the difference in statistical properties of individual

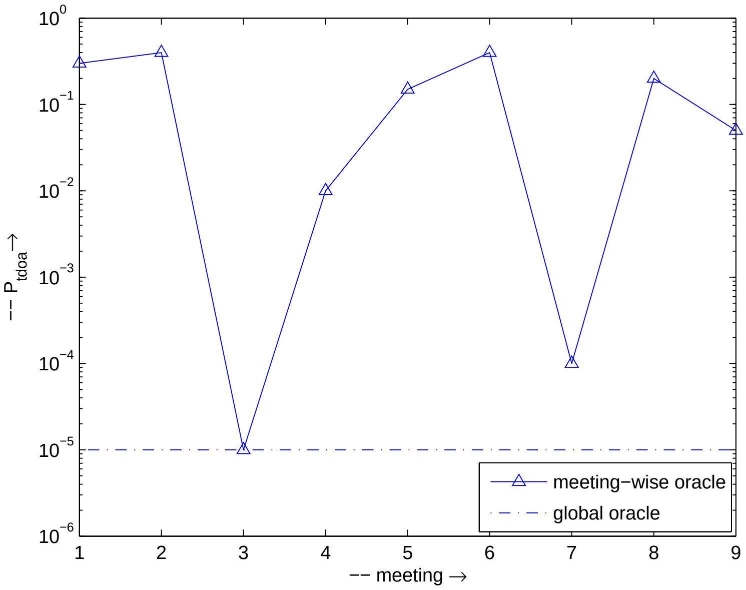


Figure 4.3. Variation of oracle weight in the HMM/GMM system for TDOA feature across different meetings $P_{mfcc} = 1 - P_{tdoa}$

feature streams as discussed before.

2. if the global weights are considerably different from the oracle values (meetings 4, 8 and 9), the drop from the optimum performances can be large.

With the weights estimated from the development data, the actual speaker error is almost double of the speaker error with oracle weights.

In the following we investigate IB system with an alternative combination scheme.

4.2 IB System Extension to Multiple Features

Let us now consider the case in which MFCC and TDOA features from the same meeting are available. The proposed method can be extended using separate aligned background GMMs for MFCC and TDOA. The background models have the same number of components proportional to the length of the meeting as described in Section 3.1.1.

Initially a GMM model is estimated using MFCC features s_t^{mfcc}. Each observation s_t^{mfcc} is then assigned to one of the GMM components. The parameters of the TDOA GMM are estimated using the same mapping between the feature time indices and the GMM components. In other words, suppose the r^{th} component parameters of MFCC GMM were estimated from a set of MFCC features $\{s_{t'}^{mfcc}\}$. The r^{th} component parameters of the TDOA GMM will then be estimated from the set of TDOA features $\{s_{t'}^{tdoa}\}$ that have the same time indices $\{t'\}$.

While in the baseline system [88] MFCC and TDOA GMM for each cluster are estimated separately with observations (MFCC and TDOA) assigned to the same cluster, in the proposed system separate background models are estimated with observations assigned to the same components (from the MFCC background model). Thus the two GMMs have the same number of components and have a strict one-to-one mapping between the components.

The set of these corresponding aligned mixture components represent the relevance variables. The relevance variable distributions $p(y|s_t^{mfcc})$ and $p(y|s_t^{tdoa})$ are estimated as before using Bayes' rule. The estimation of $p(y|s_t^{mfcc}, s_t^{tdoa})$ is obtained as a weighted average of individual distributions as:

$$p(y|s_t^{mfcc}, s_t^{tdoa}) = p(y|s_t^{mfcc})P_{mfcc} + p(y|s_t^{tdoa})P_{tdoa} \tag{4.7}$$

where P_{mfcc} and P_{tdoa} represent the weights such that $P_{mfcc} + P_{tdoa} = 1$. In contrary to GMM log-likelihood combination, here the individual distributions $p(y|s_t^{mfcc})$ and $p(y|s_t^{tdoa})$ are normalized and have the same dynamic range regardless of the dimension of the feature vector. Thus the linear combination does not suffer from dimensionality/statistics problems as in the case of GMM log-likelihoods.

4.2.1 Experiments and Results

In order to investigate the effectiveness of the proposed approach, experiments are conducted to study the combination of MFCC and TDOA features on the same set of meetings. The effect of the realignment algorithm will be discussed in the next section.

The evaluation is done on the meeting recordings described in Table 4.1. As before we report the performance of the system using meeting-wise and global oracle as well as estimated weights. The feature weights are determined using the same development data set described in Section (3.5.1).

Table 4.3 presents the speaker error values. It can be seen that the difference between global

Table 4.3. Overall speaker error for MFCC+TDOA combination of the IB system: oracle weights for each meeting and the estimated weight from separate development data ($P_{tdoa} = 0.3$). $P_{mfcc} = 1 - P_{tdoa}$

meeting-wise	global oracle	Estimated wt
oracle	$P_{tdoa} = 0.2$	$P_{tdoa} = 0.3$
8.7	10.4	11.6

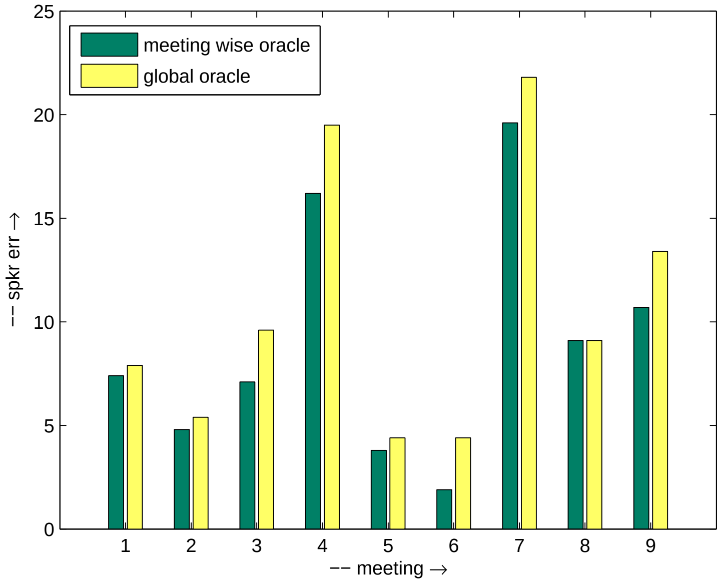


Figure 4.4. Meeting-wise speaker error for MFCC+TDOA of IB based feature combination: meeting-wise and global oracle weights for each meeting

oracle and meeting-wise oracle in the IB system (1.7%) is less compared to the HMM/GMM system (4.7%). The global oracle speaker error with aIB clustering is 1.3% absolute better than the baseline result even before performing the realignment step.

The meeting-wise speaker errors in Figure 4.4 show that the global oracle performances are close to the best performance determined by the oracle weights. The oracle weights for each meeting in case of IB system are represented in Figure 4.5 and as opposed to the HMM/GMM system (Figure 4.3), they span a smaller range. Figure 4.6 depicts the variation of speaker error with the variation of P_{tdoa} for the meeting with highest difference between oracle and estimated weight values. The IB system performance with estimated weights is closer to the performance with oracle weights as compared to the baseline system.

The estimated values of weights are $(P_{mfcc}, P_{tdoa}) = (0.7, 0.3)$. It is interesting to notice that

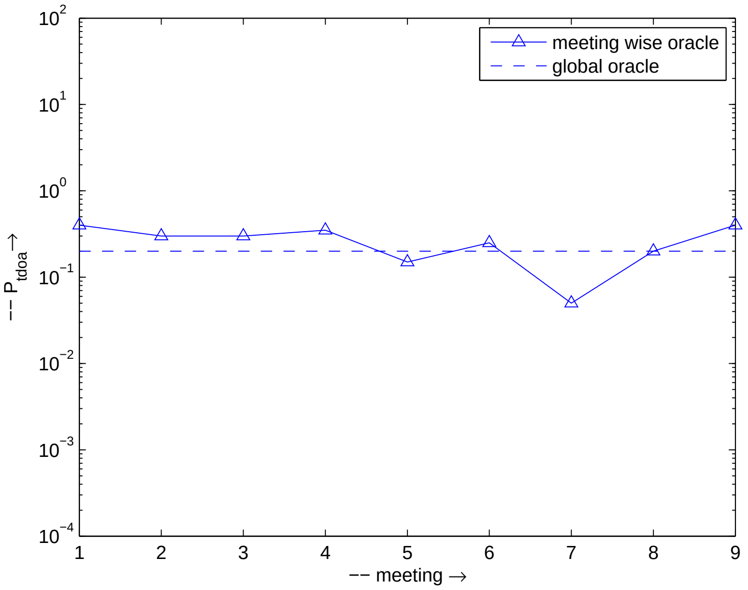


Figure 4.5. Variation of oracle weight in the IB system for TDOA features across different meetings $P_{mfcc} = 1 - P_{tdoa}$

those values are different from those obtained when the tuning is done using log-likelihood combination i.e., $(0.9, 0.1)$. The system outperform the HMM/IB system by 2% before the realignment step.

In summary, whenever the combination happens at the level of the relevance variables instead of log-likelihoods, the diarization error is less sensitive to the dimension of the TDOA features.

4.2.2 Sequential Clustering

Sequential Information Bottleneck(sIB) was proposed in Section 3.1.2 to further optimize the agglomerative clustering output. The IB objective function either improves or stays the same at each step of sIB clustering. We propose here the use of this sequential optimization on the agglomerative clustering partition. In this scenario, sIB refines the clusters by reassigning the elements similar to the cluster purification algorithms [146]. In case of multistream diarization, the distribution $p(y|x)$ calculated by Equation (4.7) can be employed. As the aIB, the sIB algorithm also requires only the distribution $p(y|x)$ as the input.

Table 4.4 reports the results in case of two and four feature streams. The sequential framework improves the performance of the oracle results as well as the performance with actual estimated

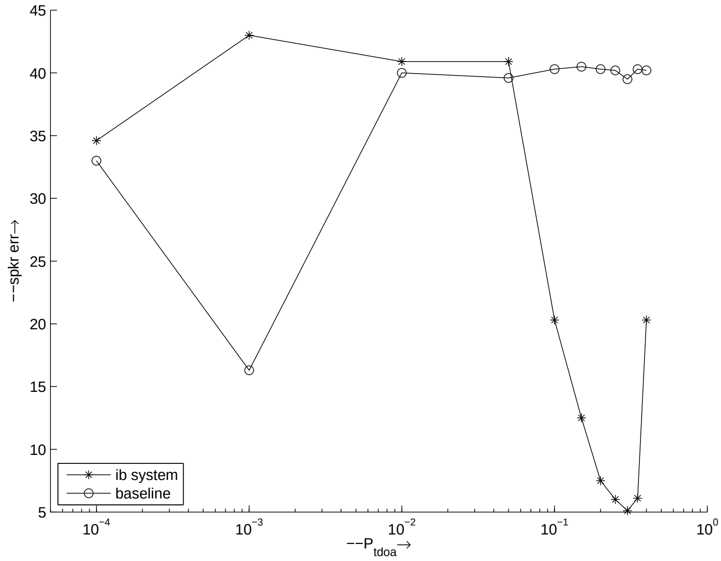


Figure 4.6. Speaker error as a function of P_{tdoa} ($P_{mfcc} = 1 - P_{tdoa}$) for a meeting with estimated weight farthest from oracle. The selected weight from development data tuning is $P_{tdoa} = 0.3$ for the IB system and $P_{tdoa} = 0.1$ for the baseline

weights. The meeting-wise and global oracle performances are improved by about 1% absolute while this improvement in case of estimated weights is 2.2%.

meeting-wise oracle	global oracle $P_{tdoa} = 0.2$	estimated wt $P_{tdoa} = 0.3$
7.50	8.60	9.40

Table 4.4. Results of sIB purification in case of meeting-wise oracle, global oracle and estimated weights.

4.3 KL based Realignment

The second contribution of this chapter consists of introducing a speaker realignment that operates directly in the space of the relevance variables estimated by Equation (3.21). The rationale is that performing realignment in the space of normalized distributions $p(Y|S)$ would increase the robustness of the system as compared to the log-likelihood domain.

4.3.1 Approach

Let us rewrite the IB objective function according to the following proposition:

Proposition 4.3.1. *The IB maximization of Equation (3.1) is equivalent to the following minimization:*

$$\min[I(X,C) + \beta\, E(d(X,C))] \tag{4.8}$$

$$d(X,C) = KL(p(Y|X)||p(Y|C)) \tag{4.9}$$

where $d(X,C)$, is the KL divergence between distributions given by the cluster and the input [46]. (See Appendix A.2 for a proof)

The re-alignment is performed after the agglomerative clustering to smooth the initial arbitrary boundaries obtained by uniform segmentation. The aIB clustering described in section 4.2 provides an initial partition of features $(s_1, \dots, s_T)$ (input variables of realignment) into a set of speakers $\{c_1, \dots, c_K\}$. This corresponds to an hard clustering partition where $p(c|s) \in \{0,1\}$. Hard clustering is obtained by taking the limit $\beta \to \infty$ in the IB optimization criterion (3.1). Thus the IB criterion reduces to the maximization of $I(C,Y)$ alone [106]. From the above proposition, this is equivalent to minimizing $d(S,C)$. Developing the expression for $d(S,C)$, it is possible to write:

$$\begin{aligned} E(d(S,C)) &= E[KL(p(Y|S)||p(Y|C))] \\ &= \sum_t p(s_t) \sum_i p(c_i|s_t) KL(p(Y|s_t)||p(Y|c_i)) \\ &= \sum_t p(s_t) KL(p(Y|s_t)||p(Y|c_t)) \end{aligned} \tag{4.10}$$

where c_t is such that $p(c_t|s_t) = 1$, for other values of C, $p(c_i|s_t) = 0$. Assuming equal priors for s_t, minimization of $E(d(S,C))$ is equivalent to:

$$\min E(d(S,C)) = \min \sum_t KL(p(Y|s_t)||p(Y|c_t)) \tag{4.11}$$

The term $p(Y|c_t)$ denotes the distribution of relevance variables for each speaker. This can be seen as the "speaker model" estimated using $p(Y|s_t)$. While the GMM realignment selects the speaker that maximizes the log-likelihood sum, the proposed approach selects the speaker that minimize the KL divergence between $p(Y|s_t)$ and $p(Y|c_t)$. The problem of minimizing the KL divergence between a feature stream represented as distributions and a set of learned models has been explored previously in the context of automatic speech recognition [13]. The estimation formula for

”speaker models” $p(y|c)$ is given by (See Appendix A.3 for a proof):

$$p(y|c_i) = \frac{1}{p(c_i)} \sum_{s_t : s_t \in c_i} p(y|s_t)p(s_t) \tag{4.12}$$

In case of equal priors $p(s_t)$, the estimation formula becomes the arithmetic mean of the distributions $p(y|s_t)$. Thus the speaker model for a cluster c_t is the average of distributions $p(y|s_t)$ assigned to it.

Then, the objective function can be extended to include the minimum duration constraint as in the baseline system:

$$\mathbf{c}^{opt} = \arg\min_{c} \sum_{t} [KL(p(Y|s_t)||p(Y|c_t)) - \log(a_{c_t c_{t+1}})] \tag{4.13}$$

A parallel can be seen between Equations (4.3) and (4.13) reported below:

$$\mathbf{c}^{opt} = \arg\min_{c} \sum_{t} [-\log b_{c_t}(s_t) - \log(a_{c_t c_{t+1}})]$$

The term $p(Y|c_t)$ represents the speaker model in the relevant variable space and during the Viterbi. The negative log-likelihood ($-\log b_{c_t}(s_t)$) is replaced by the KL divergence $KL(p(Y|s_t)||p(Y|c_t))$ which serves as the distance measure between the speaker model and the input features $p(Y|s_t)$. The realignment depends only on the distribution $p(y|s_t)$ which is normalized. When MFCC and TDOA feature streams are used, this distribution is computed as $p(y|s_t^{mfcc}, s_t^{tdoa}) = p(y|s_t^{mfcc})P_{mfcc} + p(y|s_t^{tdoa})P_{tdoa}$. Performing KL based realignment using $p(y|s_t^{mfcc}, s_t^{tdoa})$ eliminates the combination of log-likelihood scores.

4.3.2 Experiments and Results

This section compares the KL based realignment with the HMM/GMM based realignment. Both systems use a minimum duration constraint equal to 2.5 second i.e. 250 frames. The comparison is done on the same setup of section 4.1.2 after aIB clustering. Table 4.5 compares the speaker error in case of oracle and estimated weights. Figure 4.7 illustrates the meeting-wise speaker error before and after realignment.

Table 4.5. Overall speaker error for MFCC+TDOA combination of the IB system with KL realignment: using meeting-wise and global oracle weights, and estimate from development data.

	Realignment	meeting-wise oracle	global $P_{tdoa} = 0.2$	estimated wt. $P_{tdoa} = 0.3$
aIB	HMM/GMM	7.9	9.1	10.7
	HMM/KL	7.0	8.7	9.9
aIB+sIB	HMM/GMM	6.9	8.3	9.2
	HMM/KL	6.6	7.6	8.6

Table 4.6. Comparison of speaker errors of IB system (aIB and aIB+sIB systems) with the baseline. The table presents results that corresponds to weights estimated from development data.

	aIB	aIB + sIB	Baseline
no realign	11.6	9.4	13.6
HMM/GMM	10.7	9.2	
KL based	9.9	8.6	

Both realignment schemes reduce the overall speaker error in case of oracle weights as well as in case of weights estimated from development data. However the KL realignment outperforms the HMM/GMM realignment by close to 1% absolute. Figure 4.7 shows that the HMM/GMM system improves the diarization output in six out of nine meetings whereas the KL realignment is improving consistently across all meetings of the data set. All results with estimated weights are summarized in Table 4.6.

Comparing the realignment results with HMM/GMM system it can be seen that the meeting-wise oracle weights yield the same performance for the baseline system (Table 4.2) and the IB system with KL realignment (Table 4.5, Row 2). However, with the estimated set weights, performance of the IB system is ($\approx 5\%$) better as compared to the IB system. Thus the degradation from the best performance is less in case of the IB diarization system.

4.4 Summary

This chapter discussed the combination of MFCC and TDOA features for speaker diarization introducing two new contributions that extends work on information theoretic diarization in the previous chapter.

- *Combination Scheme*: State-of-the-art multiple stream diarization uses a linear combination of GMM log-likelihoods trained on MFCC and TDOA features. TDOA features have differ-

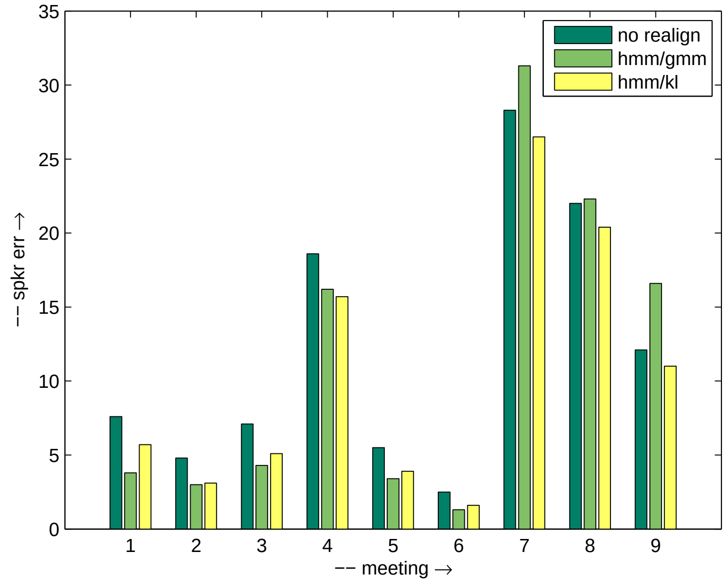


Figure 4.7. Speaker error with and without realignment for feature combination with estimated weights

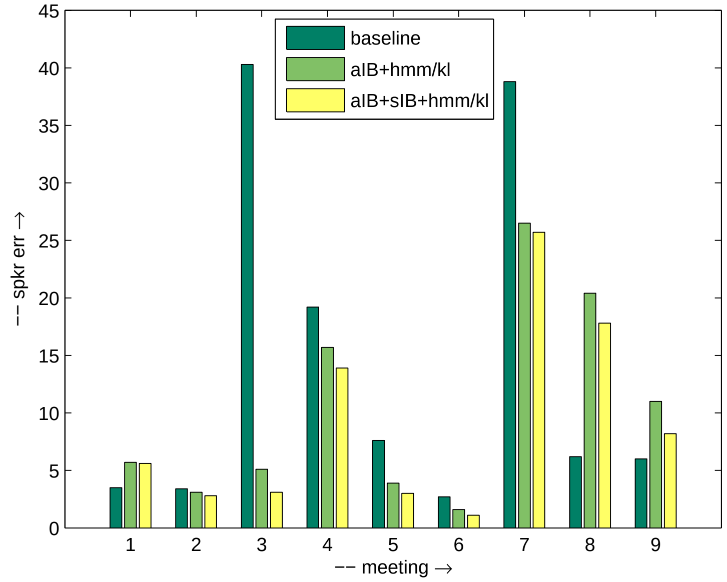


Figure 4.8. Meeting wise speaker error for the baseline system and the IB based system with KL realignment

ent statistics compared to MFCC. Furthermore their dimensionality varies according to the number of channels used for recordings. Setting linear combination weights according to log-likelihoods present robustness problems across different meeting rooms. A combination scheme performed in a normalized space of relevance variables is proposed and investigated.

- *KL based Realignment*: Instead of re-aligning boundaries with an HMM/GMM system, a KL based realignment scheme is proposed. This method uses only the frame level relevance variable distributions.

The experiments are performed on a dataset with number of TDOA features of variable dimension from 2 to 16. Both oracle weights as well as weights estimated from tuning on a development data set are investigated. The proposed combination performs 2% absolute better compared to the baseline even before realignment. Both realignments (HMM/GMM and KL) reduce the speaker error, the KL outperforming the HMM/GMM by 1% absolute. The meeting-wise results are illustrated in 4.8.

The performance of the system with aIB clustering and KL realignment is 4% absolute (28% relative) better than the baseline system. The sIB post processing further improves the output of the IB system by 1.3%. It is important to notice that the two systems hold the same oracle performance meaning that when meeting-wise oracle weights are selected, the speaker error is similar. On the other hand, whenever weights are fixed, the IB system is more robust to variations across data. The individual weights variations is much larger when the combination happens at the log-likelihood level.

Although the feature combination of only two features (MFCC and TDOA) is investigated in this work, the algorithms proposed are general and could be extended to other features (acoustic or visual). The framework only uses the distribution $p(y|s_t)$ that is normalized and is hence more robust to features with diverse statistics compared to the conventional HMM/GMM system. Experiments with more than two feature sets would be addressed in the next chapter.

Chapter 5

Diarization beyond Two Feature Streams

While TDOA stays the most common source of extra information and is implemented in all the evaluation systems, other alternatives such as modulation spectrogram features [139] or prosody based features [35] have been explored as complementary features to MFCC. However, most of the systems in literature report combination of only two feature streams, i.e., MFCC+TDOA, MFCC+modulation spectrum or MFCC+prosodic features. Though different features were found to reduce the diarization error, there was little attempt to integrate them with the MFCC+TDOA baseline and, to our best knowledge, no positive results have been reported using HMM/GMM systems.

This chapter focuses on the integration of additional feature streams together with the MFCC and TDOA in the IB based speaker diarization system. The investigation is carried with respect to two issues:

- Robustness of the diarization algorithm towards features with diverse statistics
- Estimation of relative weights of different input feature streams

The system is investigated through a set of oracle experiments as in two streams to address the first issue. We compare the performance of the log-likelihood combination with a HMM/GMM with the combination based on relevance variables in case of IB system. The actual performances are then studied using weights estimated from a development dataset. We also discuss the relative increase in computational complexity that results from incorporating additional features.

5.1 Experimental Setup

The experimental setup for evaluation consists of the same set of meetings from the RT06 evaluation campaign. The meetings' names with the number of channels that compose the microphone array are listed in Table 4.1. Multiple channels are beamformed and MFCC and TDOA features are extracted as explained in Section 4.1.2.

In addition to these features, two sets of features based on long temporal context – the Modulation Spectrum (MS) and Frequency Domain Linear Prediction(FDLP) – are explored . The rationale behind investigating those features is that they could capture the long term dependencies, which the short term spectral features are not capable of. Furthermore they have shown to provide complementary information and robustness in other meeting-related tasks like speech recognition [113], [97].

- *Modulation Spectrogram Features* – The modulation spectrum represents the slowly varying components of the spectrum. The critical band energy trajectories are filtered using a low pass filter and the resulting features are de-correlated [49]. This feature has a dimension of 26.

- *FDLP features* – Frequency domain linear prediction provides a smoothed approximation to the temporal envelope of a signal [14]. FDLP is performed over sub-bands of audio signal over a large time window (typically 1s), that yields a parametric model of the temporal envelope. Short term spectral integration is performed over smoothed envelope and short term spectral energies are converted to 19 dimensional short term cepstrum like features. A gain normalization step in the linear prediction helps to remove the artifacts from reverberrent speech [116].

In this study conventional MFCC+TDOA features (referred in the following as two streams combination) are extended to have a diarization system with MFCC + TDOA + MS + FDLP features (referred in the following as four streams combination).

To study the performances of the additional features in diarization systems independently of any weight estimation scheme or development data, two different types of oracle experiments referred as meeting-wise oracle and global oracle are investigated as before.

The meeting-wise oracle provides the best meeting-wise performance using the feature combination scheme. This is obtained by varying the individual feature weights P_l between zero and one such that $\sum_l P_l = 1$ for each meeting m and selecting the set of weight values that minimizes the speaker error. This results in a set of weights $\{P_l^m\}$ for each meeting recording. The global oracle is the best possible set of weights *across all meetings* that correspond to the least diarization error in the dataset. This is obtained by varying the individual features weights P_l between zero and one such that $\sum_l P_l = 1$. The set of values $\{P_l\}$ that minimizes the speaker error for the *entire* data set is selected. The oracle experiments on the evaluation dataset permit the study of the two systems under the ideal weighting that minimizes the speaker error. The comparison between meeting-wise and global oracles could provide insights to robustness of the individual feature weights. These studies are presented in Section 5.2.2 and Section 5.3.2.

The actual weights are estimated by minimizing the error on a separate development data set. The development dataset used for parameters tuning consists of ten meetings recorded across 5 different meeting rooms and is the same as described in 3.5.1. The development dataset contains meetings with variable number of microphones that is representative of the evaluation data set. Section 5.4 reports results obtained with estimated weights and performs comparison with results from oracle experiments.

5.2 HMM/GMM System

In this section, we compare the oracle performances of the HMM/GMM system with two as well as four input features.

5.2.1 Feature Combination

Let us consider the case in which multiple input features $\{s_t^l\}, l = 1, \ldots, M$ are available as the input features for speaker diarization. HMM/GMM systems use a separate model (GMMs) for each stream and a linear combination of likelihoods is considered in the computation of the BIC distance

Table 5.1. Speaker error for HMM/GMM system in case of two (MFCC+TDOA) and four (MFCC+TDOA+MS+FDLP) feature streams computed using meeting-wise and global oracles as well as weights estimated from development data

	meeting-wise	global
2 features	7.0	11.7
4 features	3.9	7.9

measure and in realignment. A separate GMM emission distribution $b^l_{c_k}(.)$ is estimated for each feature stream s^l. A combined log likelihood $\log L_{c_k}(s_t)$ is computed for each cluster c_k as:

$$\log L_{c_k}(s_t) = \sum_{l=1}^{M} P_l \log \left[b^l_{c_k}(s^l_t) \right] \tag{5.1}$$

where, P_l corresponds to the weight of each feature stream ($\sum_l P_l = 1$). This combined likelihood $\log L_{c_k}(.)$ replaces the log likelihood in the estimation of BIC criterion (Equation 3.22) and during the Viterbi realignment (Equation 4.3).

5.2.2 Oracle Experiments and Results

Table 5.1 presents the oracle result experiments on two as well as four feature streams for the HMM/GMM system. The meeting-wise oracle weights are depicted in Figure 5.2. The global oracle weights are given by $(P_{mfcc}, P_{tdoa}) = (0.99, 0.01)$ in case of two stream combination and $(P_{mfcc}, P_{tdoa}, P_{ms}, P_{fdlp}) = (0.84, 0.01, 0.05, 0.10)$ in case of four streams.

Table 5.1 shows that the use of four streams can improve the system performance over two streams from 7.0% to 3.9% in case of meeting-wise oracle weights and from 11.7% to 7.9% in case of global oracle weights. Furthermore when the oracle weights are chosen globally, i.e., same weights are used across all the meetings, the performances degrade by more the 4% absolute in both two and four features stream cases. The meeting-wise performance numbers are reported in Figure 5.1. As in the case of two stream feature combination, meeting-wise optimal weights span quite a large range (see Figure 5.2). This could happen as the result of large variations of likelihood values as illustrated in Figure 5.3.

The reason of this effect can be found in the combination scheme. The combination happens at log-likelihood level (see Equation 5.1) where an independent GMM is estimated for each feature stream. Figure 5.3 plots the average log-likelihoods computed using GMMs estimated from the four

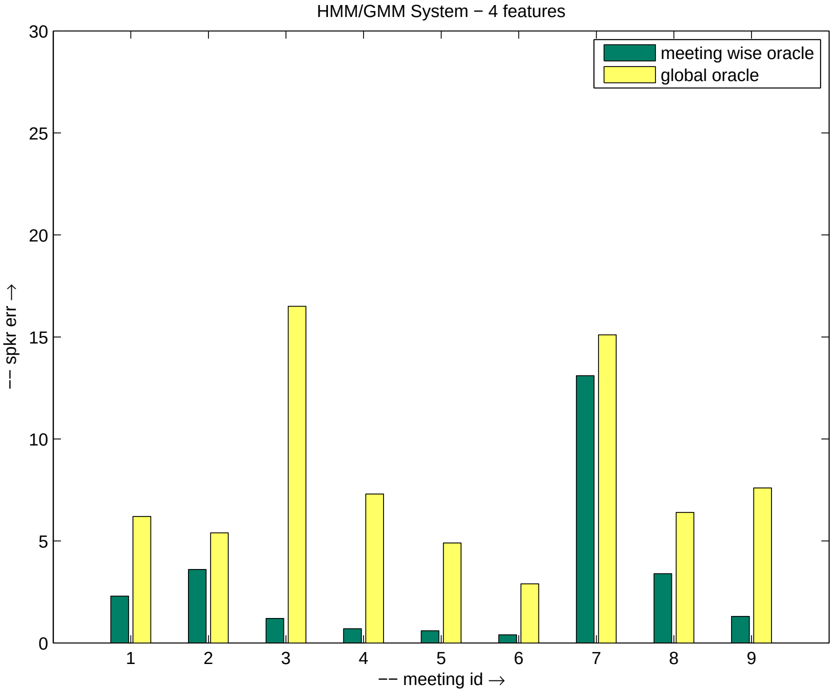


Figure 5.1. Comparison of meeting-wise oracle (feature weights that minimize speaker error for each meeting) performance with the global oracle (same feature weights across all meetings that minimize the overall speaker error) of HMM/GMM system for four stream combination.

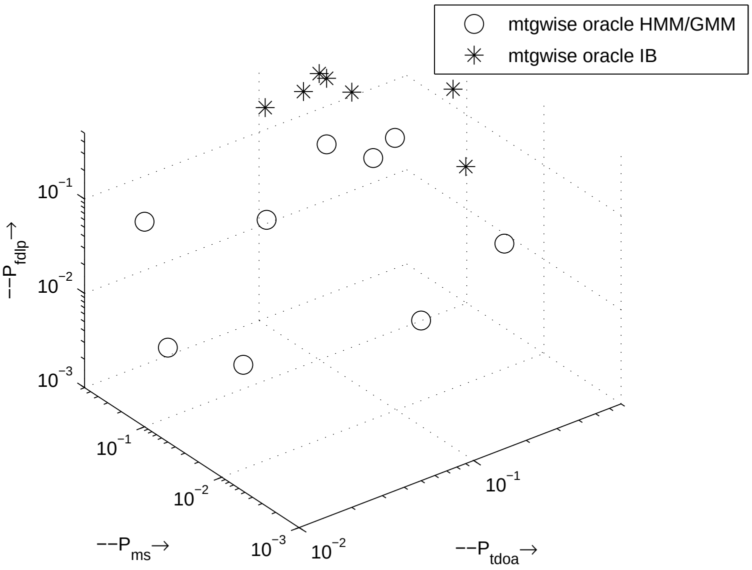


Figure 5.2. Feature weights that corresponds to lowest speaker error for each meeting (meeting-wise oracle) for the Four feature combination. The feature stream weight is varied between 0 and 1 such that $\sum_l P_l = 1$. ie., $P_{mfcc} = 1 - P_{tdoa} - P_{ms} - P_{fdlp}$

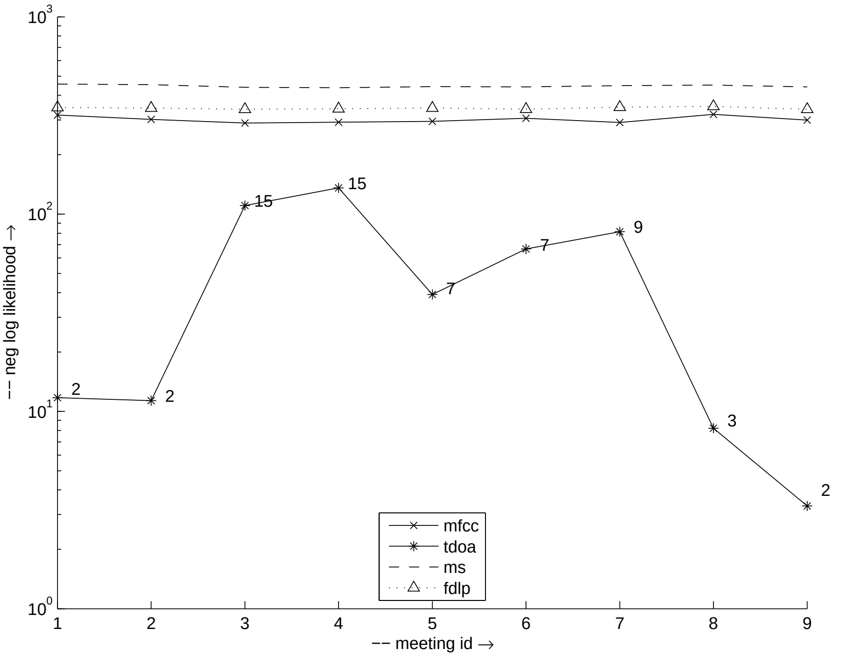


Figure 5.3. Negative log-likelihoods for different feature streams across all meetings. The numbers adjacent to the delay features denotes the feature dimension. It can be seen that the likelihood range of delay features varies across different meetings. Meetings with higher delay feature dimension (meetings 3,4) have highest value for negative log likelihoods

different streams (MFCC, TDOA, MS, FDLP). While features with constant dimensionality (MFCC, MS, FDLP) have constant log-likelihood ranges across meetings, TDOA features, whose dimensionality depends on the number of microphones, have considerably varying log-likelihood ranges. This would cause the meeting-wise oracle weights to change significantly in case of HMM/GMM (see Figure 5.2) and the performance of global-oracle degrades a lot for certain recordings (e.g. EDI meetings where arrays are composed of 16 microphones - see Figure 5.1).

5.3 IB System

In this section we perform similar oracle experiments for the IB diarization system with four feature inputs.

5.3.1 Feature Combination

The IB system performs feature combination in the relevance variable space by means of aligned background GMMs. Initially a background model is estimated for MFCC feature stream. The parameters of the other GMMs are estimated using the same mapping between feature frame indices and GMM components as described in Section 4.2. Thus there is a one-to-one correspondence between the Gaussian components of the background GMMs.

Following the estimation of the background GMMs from each feature stream $\{s_t^l\}$ $t = 1\ldots, M$, the distributions $p(y|s_t^l)$ are estimated using Bayes' rule as in Equation 3.21. The estimation of $p(y|s_t^1, \ldots, s_t^M)$ is obtained as a weighed average of individual distributions, i.e.:

$$p(y|s_t^1, \ldots, s_t^M) = \sum_l p(y|s_t^l) P_l \tag{5.2}$$

where P_l represents the weight corresponding to feature s_t^l such that $\sum_l P_l = 1$.

The distributions $p(y|s_t^1, \ldots, s_t^M)$, are averaged across all frames in a speech segment to obtain the distribution corresponding to the input speech segment $p(y|x)$ which is then used as the input to aIB clustering. The realignment also uses the combined distribution $p(y|s_t^1, \ldots, s_t^M)$.

5.3.2 Oracle Experiments and Results

As before, we consider the meeting-wise and global oracle performance of the algorithm. The results are reported in Table 5.2. Results show that the use of four streams can improve the diarization error as compared to two streams from 7.0% to 3.4% in case of meeting-wise oracle weights and from 8.7% to 5.7% in case of global oracle weights. The meeting-wise oracle weights are depicted in Figure 5.2. The global oracle weights are given by $(P_{mfcc}, P_{tdoa}) = (0.80, 0.20)$ in case of two stream combination and $(P_{mfcc}, P_{tdoa}, P_{ms}, P_{fdlp}) = (0.45, 0.15, 0.15, 0.25)$ in case of four streams. When the oracle weights are chosen globally, i.e., with same weights for all the meetings, the performances degrade by 2% in both two and four feature stream case. Comparing the IB system with the HMM/GMM, it is possible to notice that,

- The meeting-wise oracle results are similar for the two systems.

- When the meeting-wise oracle weights are replaced with global weights the HMM/GMM sys-

Table 5.2. Speaker error for IB system with KL realignment in case of two (MFCC+TDOA) and four (MFCC+TDOA+MS+FDLP) stream combinations computed using meeting-wise and global oracle weights

	meeting-wise	global
2 features	7.0	8.7
4 features	3.4	5.7

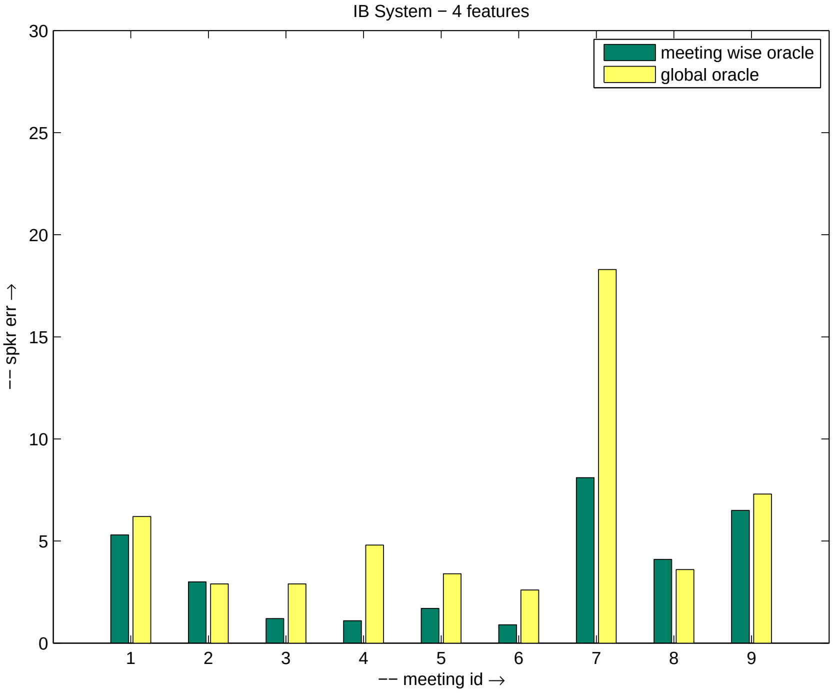


Figure 5.4. Comparison of meeting-wise oracle (feature weights that minimize speaker error for each meeting) performance with the global oracle (same feature weights across all meetings that minimize the overall speaker error) for four feature stream combination with IB system

tem degrades by 4% absolute while the IB system degrades only by 2%. This is verified both in case of two and four stream systems.

In contrary to HMM/GMM system, the best per-meeting weights are in similar range in case of the IB system. Furthermore Figure 5.4 plots the speaker error for all the meetings when weights are chosen by meeting-wise oracle and by global oracle. It can be noticed that in case of HMM/GMM, the use of a global weights instead of meeting-wise weights degrades the performance considerably as compared to the IB system in certain cases (meeting ids 3, 4, 5, 9).

In case of IB feature combination system, the feature combination happens in the space of relevance variables. Since individual distributions are normalized, problems associated with diverse statistics of features are avoided. This explains that meeting-wise oracle weights have similar

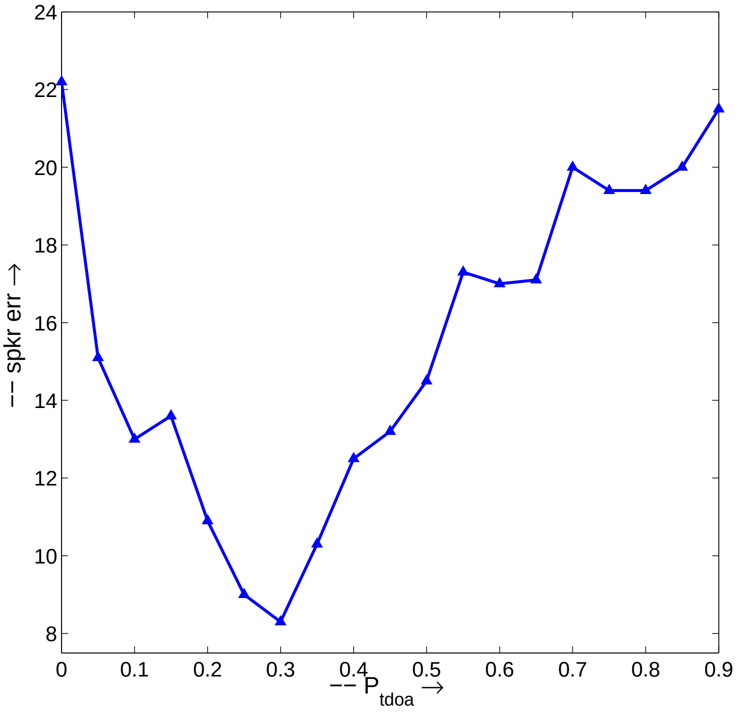


Figure 5.5. Speaker error of IB system as a function of feature stream weight P_{tdoa} (2 feature stream). the global minimum is $(P_{mfcc}, P_{tdoa}) = (0.7, 0.3)$ $P_{mfcc} = 1 - P_{tdoa}$

ranges (see Figure 5.2) and that the performance of global-oracle degrades comparatively less w.r.t. the HMM/GMM system (see Figure 5.4).

5.4 Estimated combination weights

Let us now consider the case in which the combination weights are estimated from the development data set described in section 5.1. The weights are selected based on the set of weights that achieve minimum error in the development dataset.

Consider the speaker error in the development data as a function of the stream weights in the two as well as four stream combination. Since, the four weights $(P_{mfcc}, P_{tdoa}, P_{ms}, P_{fdlp})$ lie in a three dimensional subspace ($\sum_l P_l = 1$), the speaker error can be visualized by fixing one of the weights and plotting as a function of the other two. Figure 5.7 and Figure 5.8 illustrate the speaker error function in the development data for the IB and HMM/GMM systems respectively. While in

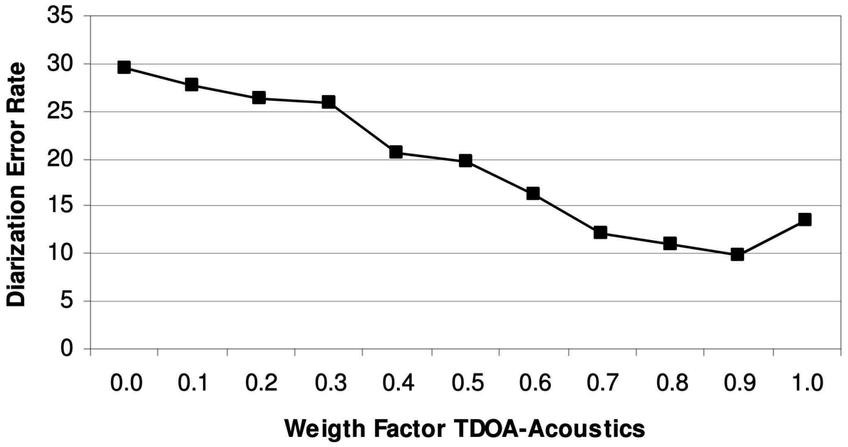


Figure 5.6. Speaker error of HMM/GMM system as a function of stream weight P_{mfcc} (2 feature stream). the global minimum is $(P_{mfcc}, P_{tdoa}) = (0.9, 0.1)$ $P_{tdoa} = 1 - P_{mfcc}$ (90)

case of two feature streams (MFCC and TDOA) the problem has just one degree of freedom and the speaker error is a smooth function and with a well defined minimum (Figure 5.5, 5.6), in case of four features the problem has three degrees of freedom that has no well defined minimum.

In order to increase the robustness of the weight estimation in case of four streams, a simple smoothing procedure is proposed. The original error is convolved with a multi-dimensional Gaussian given by:

$$\begin{aligned} g[l,m,n] &= e^{-\frac{1}{2\sigma^2}(l^2+m^2+n^2)}; |l|,|m|,|n| \leq 1 \\ &= 0; \text{otherwise} \end{aligned} \tag{5.3}$$

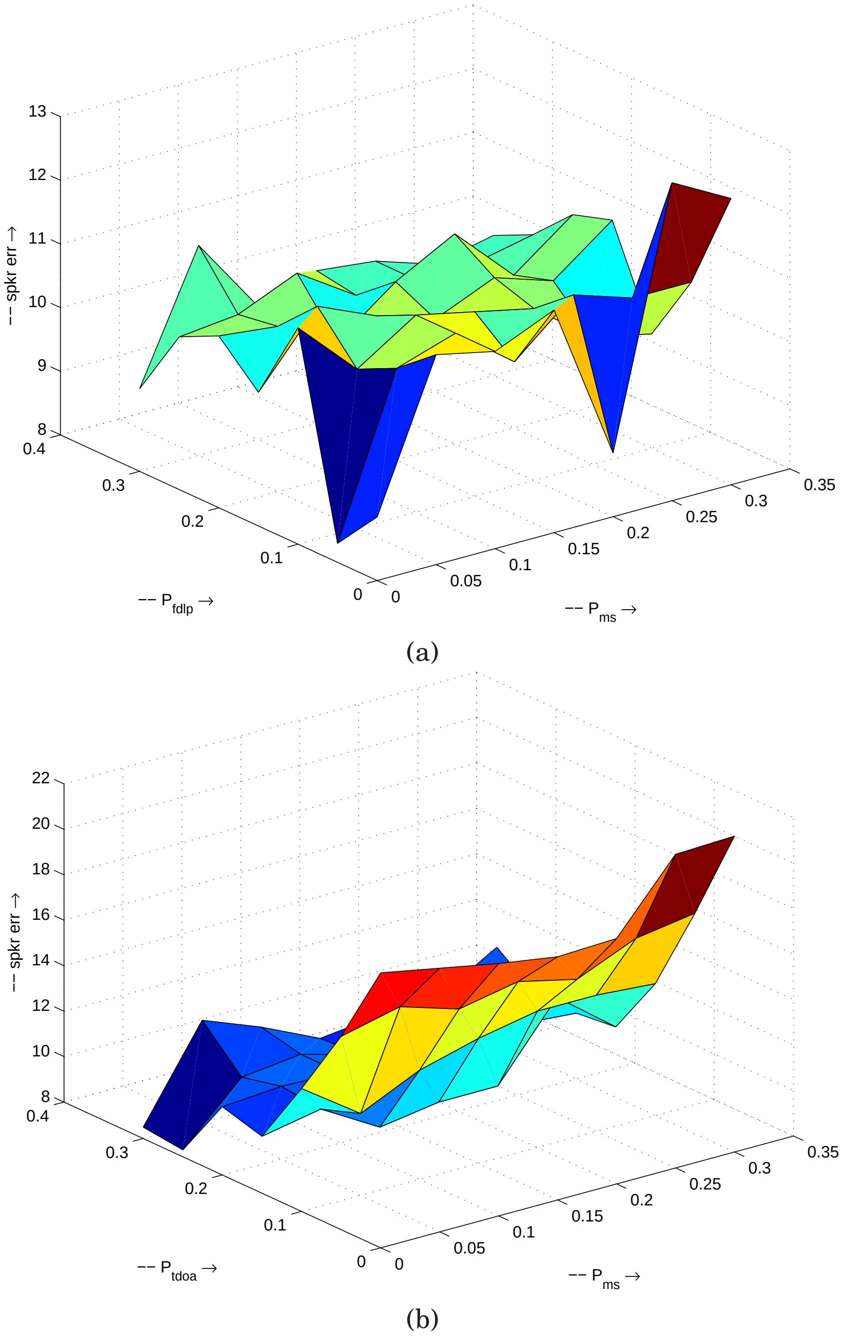
The filter is a low pass filter centered at the origin. It computes the weighted average of the center point with the 26 neighboring points. The weights are then chosen as the point with lowest value for the filtered speaker error function. The rationale behind this, is to avoid local minima of the function that does not generalize well on the test data.

5.4.1 Experiments and Results

The speaker errors are reported in Table 5.3

Results show that, whenever weights are selected according to a development data set:

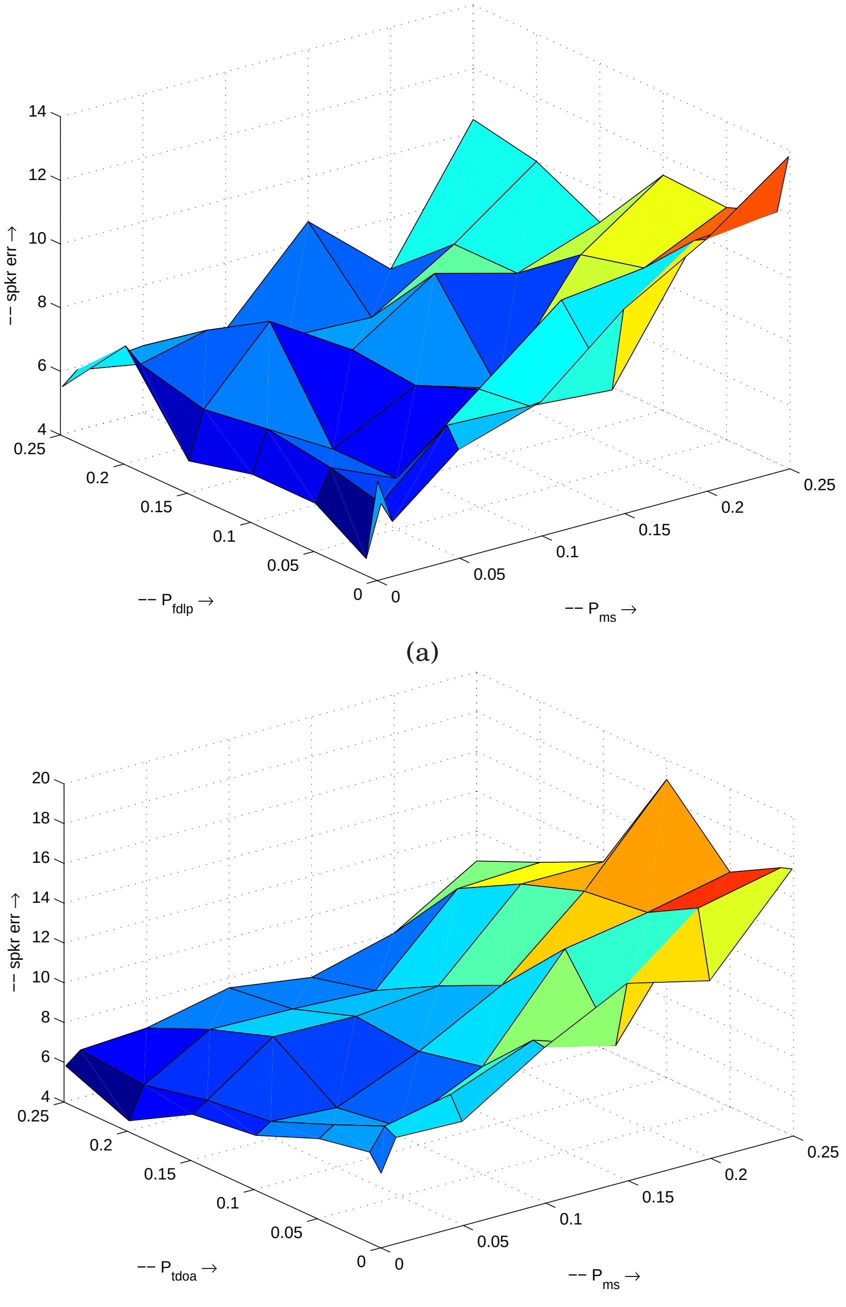
- The filtering does not alter the weights in case of MFCC+TDOA combination and hence the



(a)

(b)

Figure 5.7. Speaker error of IB system as a function of feature stream weights (four feature stream). The global minimum is $(P_{mfcc}, P_{tdoa}, P_{ms}, P_{fdlp}) = (0.7, 0.25, 0, 0.05)$ (a) Fixed $P_{tdoa} = 0.25$ and (b) Fixed $P_{fdlp} = 0.05$. $P_{mfcc} = 1 - (P_{tdoa} + P_{ms} + P_{fdlp})$



(a)

(b)

Figure 5.8. Speaker error of HMM/GMM system as a function of feature stream weights (four feature stream). The global minimum is $(P_{mfcc}, P_{tdoa}, P_{ms}, P_{fdlp}) = (0.74, 0.15, 0.01, 0.10)$ (a) Fixed $P_{tdoa} = 0.15$ and (b) Fixed $P_{fdlp} = 0.10$. $P_{mfcc} = 1 - (P_{tdoa} + P_{ms} + P_{fdlp})$

Table 5.3. Comparison of speaker error : Performance of the HMM/GMM system and IB system with estimated weights from development data using minimum value of the filtered speaker error

	HMM/GMM	aIB
2 features	13.6	9.9
4 features	14.5	6.7

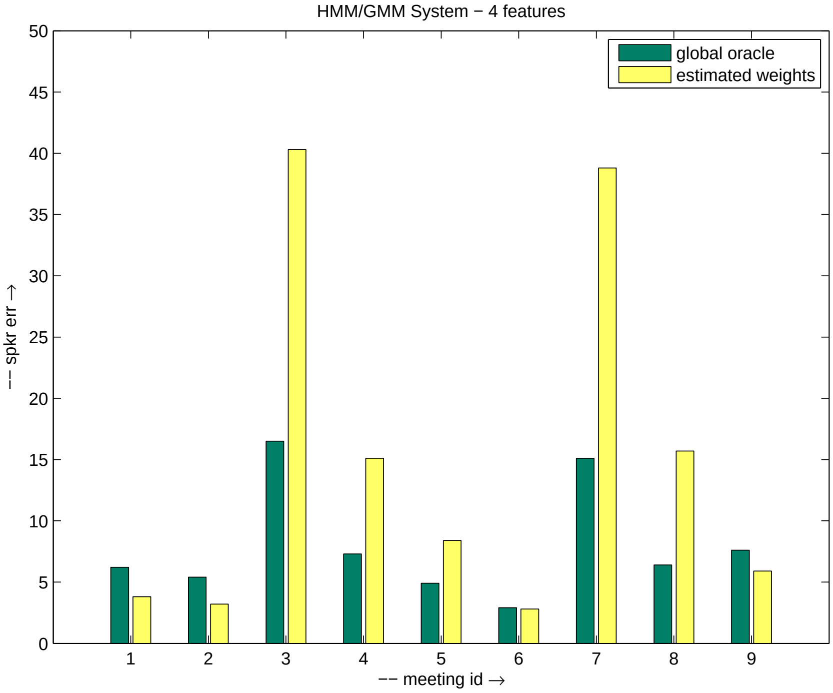


Figure 5.9. Comparison of meeting-wise performances of global oracle and estimated weights for four feature stream combination with HMM system

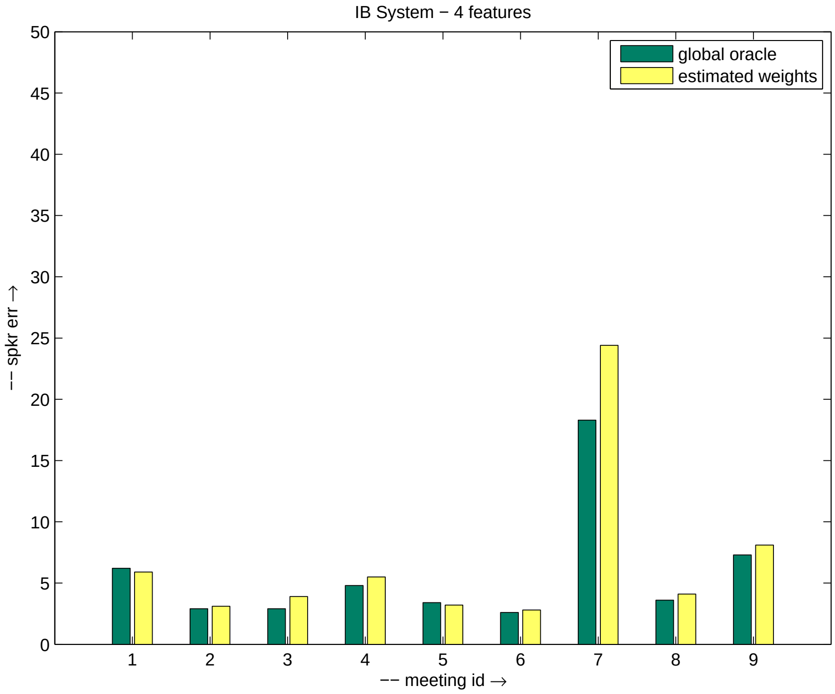


Figure 5.10. Comparison of meeting-wise performances of global oracle and estimated weights for four feature stream combination with IB system

two stream results remain constant.

- Both system performances degrade as compared to the global oracle, the difference being larger in HMM/GMM system ($\approx 6.5\%$) as compared to IB system ($\approx 1\%$) in case of four stream features.
- The HMM/GMM system performance with four streams do not improve as compared to the two stream in spite of the increase in oracle performance.
- The IB system benefits from the four stream combination and the system performance improves by 3% as compared to two stream combination.
- The IB outperforms the HMM/GMM system, the difference being 3.3% in case of two features and 7% in case of four features.

Comparison of the meeting-wise results with estimated weights and global oracle weights are presented in Figure 5.9,Figure 5.10 for HMM/GMM and IB systems respectively. In case of HMM/GMM system, the degradation in performance with estimated weights as compared to oracle is higher in

Table 5.4. Comparison of speaker error : selecting the feature stream weights based on minimum value of speaker error Vs minimum value of filtered speaker error

	aIB	aIB + sIB
2 features	9.9	8.6
4 features	6.7	6.0

meetings with higher dimension of TDOA features (meetings 3,4 7). This degradation is more uniform in case of IB system.

5.4.2 Sequential IB Purification

As discussed in the Section 3.4.3 it is possible to perform a sequential clustering to refine the output of aIB clustering system. Table 5.4 reports the results in case of two and four feature streams. The sequential framework improves the performance by 1.3% absolute in the first case (from 9.9% to 8.6%) and by 0.7% absolute (10% relative) in the second case (from 6.7% to 6.0%).

The improvement obtained by the sequential optimization is small while the number of streams increases and the speaker error becomes very low.

5.5 Computational Complexity

This section analyzes the complexity of the HMM/GMM and IB diarization systems. Both systems follow an agglomerative clustering framework. This requires calculation of distance measure between every pair of clusters. In case of HMM/GMM system this distance measure is given by the BIC criterion. Computing this involves the estimation of a Gaussian Mixture Model(GMM) using the Expectation Maximization algorithm. In case of multiple feature streams a new GMM model needs to be estimated for each feature stream per each BIC computation (see Equation 5.1), thus increasing the computational complexity.

In contrast to this, the IB system estimates a background GMM only once and then the combination happens in the space of relevance variable distributions (Equation 5.2). The distance measure in this space is based on a Jensen-Shannon divergence (Equation 3.7) that can be computed in close form. This calculation does not depend on the number of feature streams, since the dimension of relevance variables depends only on the number of components of the background GMM models.

Table 5.5. Algorithm time used by different steps - relevance distribution estimation, agglomerative clustering, KL realignment - in terms of real time factors for the IB system

	aIB			aIB+sIB		
	estimate $p(y\|x)$	IB clstrng	KL realgn	estimate $p(y\|x)$	IB clstrng	KL realgn
2 feat	0.24	0.08	0.09	0.25	0.10	0.09
4 feat	0.52	0.09	0.11	0.52	0.10	0.11

Table 5.6. Comparison of real time factors between IB and HMM/GMM system

	Baseline	aIB	aIB+sIB
2 feats	3.8	0.41	0.43
4 feats	11.3	0.72	0.75

Thus, once the combined distribution $p(y|x)$ is estimated, computational complexities of clustering and realignment steps stay the same.

The algorithm complexities are bench-marked on a normal desktop Machine (AMD Athlon™ 64x2 Dual core processor 2.6 GHz, 2GB RAM). The run-time of algorithms are averaged across different meetings and multiple iterations. Table 5.5 reports the real time factors of various steps of IB diarization algorithm. Majority of the algorithm time (roughly 60% in case of two features and 70% in case of four features) is spent in estimating the relevance variable distributions. The clustering and realignment complexities stay almost constant with the addition of new features. In both cases, the additional complexity introduced by sIB step is minimal ($\approx 12\%$ of clustering time).

Table 5.6 illustrates the comparison with HMM/GMM system. The proposed IB system runs in real time and is 8 times faster than HMM/GMM system in the two stream case and around 14 times faster in the four stream case. Addition of two feature streams slows down the HMM/GMM system by a factor of 3, while this is only 1.7 in case of IB system. Thus the IB system is computationally more efficient as compared to the HMM/GMM.

5.6 Summary

Speaker diarization based on multiple feature streams has been an active field of research during the recent years. While the most studied and used combination consists of MFCC and TDOA feature, no success has been reported in literature on integrating other feature streams together with those.

This chapter investigates the use of two features with long temporal context, i.e. the Modulation Spectrum and FDLP together with MFCC and TDOA. The study is done on a set of 9 meetings recorded across five meeting rooms used in NIST Rich Transcription '06 campaign. In this chapter, we perform a comparison between a conventional HMM/GMM diarization system based on a modified BIC criterion and the Information Bottleneck system. The first performs a combination based on log-likelihoods while the second operates in a space of relevance variables.

The comparison is performed using both oracle weights (i.e. the weights that produces the lowest possible speaker error) and weights estimated using a development data set. Experiments performed based on oracle weights reveal that:

- Four feature streams with oracle weights reduce the speaker error in both systems compared to the two stream case by almost 50% relative.

- Both systems have comparable performance with meeting-wise oracle weights. While the weights are fixed across the meetings (global oracle), the IB system performance is superior to the HMM/GMM system. This happens since optimal meeting-wise weights for the HMM/GMM system span a larger range as it depends on the GMM log-likelihood values that change according to the number of microphones in the array. On the other hand, the IB system uses a normalized space of relevance variables thus being less sensitive to the different feature vectors dimensions.

- as a consequence of the dependency to the feature vector dimension, the HMM/GMM appears to be more sensitive to changes in individual stream weights as compared to the IB system.

Experiments based on weights estimated from a development data set show that:

- The speaker error function of stream weights in case of four streams is a highly non-smooth function with no clear minimum. In order to avoid the local minima in the development data, a simple smoothing function is applied on the development data error function of the weights. Although very simple, the filter produces considerable improvements in case of four feature streams and leave unchanged the results in case of two feature streams.

- IB results in case of four features are superior to those of the HMM/GMM which is affected by the dimensionality of the TDOA features.

- The IB system with sIB post processing improves the performance by 10% relative. The improvement is relatively small as compared to the improvements in the two stream system.

The computational requirements of HMM/GMM and IB systems are compared in terms of real time factors for the two stream as well as four stream feature combination. The two systems fundamentally differ in terms of the distance calculations. The experimental outcomes report that:

- The IB system is significantly faster (8 times with two feature streams and 14 times with four feature streams) than the HMM/GMM system and is able to achieve realtime diarization even with four input feature streams. This difference happens since the distance measure computation in case of HMM/GMM system requires estimation of a new GMM model that is computationally expensive while the IB system has a distance measure based on Jensen-Shannon divergence that is straightforward to compute.
- The realtime factor of HMM/GMM increases by a factor of 3 with the introduction of two additional features, while this factor for IB system is only 1.7. This is due to the fact that introduction of new features in the HMM/GMM system involves building separate models for each input feature stream whenever two clusters are merged. In contrary, the increase in complexity in IB system happens only in the estimation of relevance variable distributions. Time complexity of clustering and realignment algorithms of of IB based diarization system depends only on the number of relevance variables and not on number and dimension of input features.

Thus the IB system provides a robust method to integrate multiple features for speaker diarization with limited increase in computation complexity.

Chapter 6

Performance on NIST Evaluation Data

All the algorithms were so far evaluated and compared on the RT06 evaluation data set. In this chapter we perform the diarization experiments on a larger dataset that consists of meetings across different NIST diarization evaluations.

6.1 Experiments and Results

All experiments are performed in a dataset that consists of NIST "meeting diarization evaluation" meetings from the years 2006,2007 and 2009. There are 24 meetings in the dataset recorded across 6 meeting rooms. The set of meetings is listed in Table 6.1. The total length of audio recording is about 9 hours. The number of channels from meetings vary from 2 to 16. All meetings are beamformed using the BeamformIt [147] toolkit that performs a delay and sum beamforming. The beamformed audio is used to extract MFCC features, modulation spectrum and the FDLP features.

Experiments are conducted using the HMM/GMM system as well as the proposed IB diarization systems. Both the two feature combination (MFCC + TDOA) and fourstream combination are performed. The oracle performance computation involves diarization of each meeting with an exhaustive list of weights. With large number of meetings, this is computationally expensive in case of HMM/GMM, and hence only the system performance with estimated weights from development

Table 6.1. List of meeting used for evaluation associated number of microphones

sl.no.	meeting id	#microphones
1	CMU_20050912-0900	2
2	CMU_20050914-0900	2
3	CMU_20061115-1030	3
4	CMU_20061115-1530	3
5	EDI_20050216-1051	16
6	EDI_20050218-0900	16
7	EDI_20061113-1500	16
8	EDI_20061114-1500	16
9	EDI_20071128-1000	8
10	EDI_20071128-1500	8
11	IDI_20090128-1600	8
12	IDI_20090129-1000	8
13	NIST_20051024-0930	7
14	NIST_20051102-1323	7
15	NIST_20051104-1515	7
16	NIST_20060216-1347	7
17	NIST_20080201-1405	7
18	NIST_20080227-1501	7
19	NIST_20080307-0955	7
20	TNO_20041103-1130	9
21	VT_20050408-1500	4
22	VT_20050425-1000	7
23	VT_20050623-1400	4
24	VT_20051027-1400	3

data are reported. The same sets of weights as reported in Chapter 4 and 5 are employed for the combination. We use a common speech-non speech segmentation with a missed speech error of 7.3% and false alarm rate of 0.4% across all experiments. Thus the speech non-speech error is fixed at 7.7% and only speaker error is reported henceforth.

Table 6.2 reports the results for HMM/GMM system as well as IB system (with and without sIB purification). It can be seen that the IB system performs $\approx 2\%$ better than the HMM/GMM system in case of two feature streams. However the HMM/GMM system do not benefit from the additional features as observed in Chapter 5. At the same time, the IB system performance improve by 5%

Table 6.2. Comparison of speaker error for the evaluation – HMM/GMM and IB diarization system performances for two streams (MFCC with TDOA features) and four stream (MFCC, TDOA, Modulation Spectrum and FDLP features) inputs

		IB	
	HMM/GMM	aIB	aIB+sIB
2 stream	14.3	12.3	11.3
4 stream	14.5	7.2	6.8

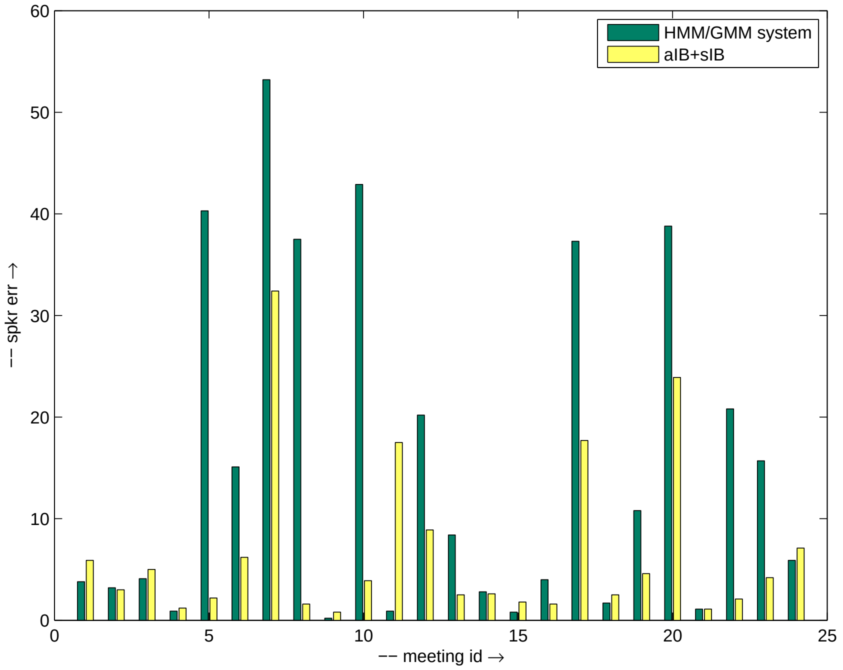


Figure 6.1. Comparison of HMM/GMM and IB system performances in case of four stream combination

absolute in case of four feature stream combination. These results are similar to the NIST RT'06 Evaluation performance reported in Table 4.6. Sequential IB algorithm improves the performance further by around $\approx 1\%$ absolute for two feature stream. The improvement in case of four feature streams is very small (only 0.4%). The meeting-wise comparison of performances for the four stream systems are presented in Figure 6.1. It can be seen that the meetings where IB system outperform the HMM/GMM system have higher number of microphones (meetings 5,6,7,8,10, 20). The delay features have a higher dimension in these meetings that leads to a higher log-likelihood range. This could affect the performance of the HMM/GMM system since it depends on linear combination of log-likelihoods.

6.2 Conclusions

In this chapter, we evaluated the proposed system on a larger dataset. Experiments reveal that while the HMM/GMM system is not able to utilize the four feature streams, IB system performance improves considerably. The IB based diarization system has a speaker error that is half as compared to the HMM/GMM system.

Chapter 7

Summary and Conclusions

Speaker Diarization is a crucial step in many meeting analysis tasks. In this domain, there is need for fast diarization systems with low computational complexity for some applications like meeting summarization and analysis while the meeting is taking place. This would enable realization of several applications (meeting browsing, speaker retrieval) on normal desktop machines. However, conventional diarization systems depend on parametric models to compute the BIC distance measure for diarization and require the re-estimation of parametric models for every possible cluster pair merge. This is computationally very expensive and require considerable optimization to make it feasible for real time speaker diarization.

7.1 IB Based Diarization

In this thesis, we propose a distributional clustering algorithm for speaker diarization based on Information Bottleneck (IB) principle [131]. The algorithm aims to minimize the mutual information loss with respect to a set of relevance variables, through which the knowledge about the problem is introduced. The method depends only on the distribution of those relevance variables and avoids estimation of speaker models. We follow an agglomerative optimization of the Information Bottleneck criterion and propose a model selection criterion to determine the number of speakers. While evaluated on the NIST RT06 meeting diarization task, the performance of the IB system (17.1%) is similar to a state-of-the-art HMM/GMM system (17.0%) with a considerable improvement in diarization time ($0.22 \times RT$) as compared to the HMM/GMM system ($3.5 \times RT$).

One drawback in agglomerative algorithms is the convergence to sub-optimal solutions due to greedy natures. Purification algorithms aim to reassign the speech frames that are assigned to the wrong class. We also present a purification scheme based on sequential optimization of the same objective function to improve the agglomerative clustering output [132]. The sequential optimization improves the output of agglomerative algorithm (16.6%). The increase in computational complexity by adding the sequential optimization is very small and the overall system perform in $0.25 \times RT$ [134].

7.2 Multiple Input Features

More recently speaker diarization systems tend to exploit multiple feature stream inputs to further improve their performance. Towards this goal conventional systems build separate speaker models for each feature stream. A linear combination of log likelihoods is then employed in the distance computation and realignment steps. However, different input features might possess different statistical properties. For example the dimension of TDOA features depends on the number of microphones, which may vary across meetings. In addition, the log-likelihood range of each meeting might possess different dynamic range.

7.2.1 Feature combination for IB clustering

We propose a feature combination scheme in the space of relevance variables [133]. Distribution of the relevance variables are estimated from each feature stream. The feature combination is performed as the linear combination of individual distributions. The combined distribution is then used as the input to agglomerative clustering. Since the individual distributions are normalized, issues related to log-likelihood combination are avoided. The algorithms are evaluated on RT06 Evaluation data with the conventional MFCC and TDOA combination. Oracle experiments reveal that the ideal weights for feature combinations span a large range in case of log-likelihood combination. On the other hand, in case of distribution based combination, the weights have a much smaller range. Using the actual estimated weights from development data, the IB system achieves a speaker error of 11.6% which represents an absolute improvement of 2%. over the HMM/GMM

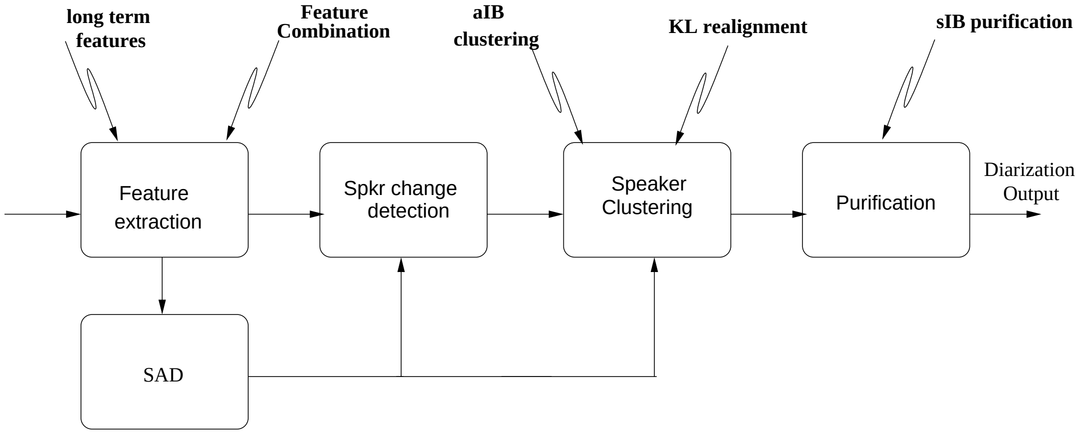


Figure 7.1. Various contributions of this thesis in the diarization system (represented in bold)

system.

7.2.2 Viterbi realignment

Speaker diarization systems make an extensive use of Viterbi realignment algorithm for incorporating the minimum duration constraint and smooth the speaker boundaries. An ergodic HMM with minimum duration constraint for each speaker is employed for this purpose. We develop a realignment algorithm that optimizes a special case of IB criterion with a minimum duration constraint [135]. The realignment algorithm also depends only on the relevance variable distribution. This eliminates the need for explicit log likelihood combination from realignment step. Experiments show that the proposed Viterbi realignment algorithm yields similar results as HMM/GMM realignment in case of single stream diarization. However, the proposed realignment performs better than HMM/GMM realignment in case of multiple streams (From 10.7% to 9.9%) [138].

7.2.3 Beyond two streams

Different features including TDOA, modulation spectrum and prosody based features have been studied as complementary features for MFCC. While most of these features improve the MFCC alone system, most state of the art systems use a combination of MFCC and TDOA features only. There are few attempts to combine more than these two features to improve the MFCC+TDOA performance. We extend the proposed system to more than two features by the addition of two more feature sets – Modulation Spectrogram and Frequency Domain Linear Prediction – with MFCC and TDOA combination and investigate the robustness of the algorithms [137]. Oracle experiments show improvements in case of both HMM/GMM system as well as IB systems. However, with estimated weights the performance of the HMM/GMM degrades as compared to the MFCC and TDOA combination. In contrary, the IB system is able to achieve a significant error reduction (2.2% absolute) over the two stream combination. To our best knowledge, this is the first successful attempt to combine more than two features for diarization. Figure 7.1 summarizes various propositions made in this thesis.

Analysis of algorithm run time on RT06 Evaluation data shows that the only increase in computational complexity that happens with additional features is in estimation of the relevance variable distributions. The clustering and realignment algorithms depend only on the distribution, and hence their complexity remains the same once the distributions are computed. The system run time is less than realtime both with two streams ($0.43 \times RT$) as well as four streams ($0.75 \times RT$) [136].

7.3 Future Directions

A number of future directions can be foreseen to further improve the current system. In between those, let us mention:

- **Universal background model:** Currently, the set of relevance variables are estimated from the background GMM, that is estimated from the same meeting. The amount of available data is then relatively small. A Universal Background Model (UBM) trained from large number of speakers was employed in the context of speaker recognition systems [93]. In a similar manner, a UBM could be used to replace the background GMM in IB diarization. This could further improve the system. In addition, the current scheme of using a background GMM from the same meeting require the entire meeting recording before starting the diarization. This constraint will be eliminated by the UBM that would eventually enable on-line diarization.

- **Overlapping speech:**The current system always assigns a speech segment to one particular speaker and does not consider the case when two or more speakers talk together (overlapping speech). Speech data in case of meeting rooms contains considerable amount of overlapping speech, and it was shown that overlap detection can improve the diarization performance (Eg: [19]). The method performs an overlap detection, and perform the speaker clustering on segments that are not overlapping. Incorporating a similar overlap detection/segmentation module can help to process the overlapping speech separately, thus improving the diarization.

- **Automatic feature weights:** Whenever multiple input features are present, the IB diarization system requires a development dataset tuning to estimate the feature weights. Alternative methods such as combination of posterior feature streams with automatic weighting has been explored in the context of ASR. (Eg. [72]). This could eliminate the dependency of feature combination on the development data. Although some attempts in this direction have been done, e.g. [142], the problem of automatically selecting the feature weights is largely unexplored.

- **Multimodal diarization** Multimodal diarization is an emerging paradigm (See [34] for an example). Video features such as visual focus of attention and intensity of motion to contain diarization information that are complementary to spectral based features [37]. Thus whenever, synchronized video and audio data are available for a meeting recording, the current system could benefit from incorporating such visual features.

Appendix A

Appendices

A.1 MDL criterion for IB clustering

The optimal model minimizes the following criterion.

$$\mathcal{F}_{MDL}(m) = L(m) + L(X|m) \tag{A.1}$$

where $L(m)$ is the code length to encode the model with a fixed length code and $L(X|m)$ is the code length required to encode the data given the model.

To determine $L(X|m)$ let us consider N input samples clustered into M clusters. The average number of input samples per cluster is $\frac{N}{M}$. Therefore the point in each cluster can be modeled with a code of length $\log \frac{N}{M}$. The total number of bits for all the N input samples is given by:

$$L(X|m) = N \log \frac{N}{M} \tag{A.2}$$

$L(m)$ represents the optimal number of bits to encode the clusters with the relevance variables is given by the entropy $H(Y, C)$ that can be written as:

$$(H(Y|C) + H(C)) \tag{A.3}$$

Since $H(Y|C) = H(Y) - I(Y,C)$ the MDL criterion becomes:

$$\mathcal{F}_{MDL} = N[H(Y) - I(Y,C) + H(C)] + N \log \frac{N}{M} \tag{A.4}$$

A.2 Alternate form of IB Objective function

Here we prove the equivalence in Proposition 4.3.1 which states:

$$\arg\max I(C,Y) - \frac{1}{\beta} I(X,C) = \arg\min \left[I(X,C) + \beta \sum_{x,c} p(x,c) KL\left(p(Y|x)||p(Y|c)\right) \right]$$

Consider the $I(X,Y) - I(C,Y)$

$$\begin{aligned}
&= \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} - \sum_{y,c} p(y,c) \log \frac{p(y,c)}{p(y)p(c)} \\
&= \sum_{x,y,c} p(x,y,c) \log \frac{p(x,y)p(c)}{p(y,c)p(x)} \\
&= \sum_{x,y,c} p(y|x,c)p(c|x)p(x) \log \frac{p(x,y)p(c)}{p(y,c)p(x)} \\
&= \sum_{x,y,c} p(y|x)p(c|x)p(x) \log \frac{p(y|x)}{p(y|c)} \\
&= \sum_{x} p(x) \sum_{c} p(c|x) \sum_{y} p(y|x) \log \frac{p(y|x)}{p(y|c)} \\
&= \sum_{x} p(x) \sum_{c} p(c|x) KL\left(p(Y|x)||p(Y|c)\right) \\
&= \sum_{x,c} p(x,c) KL\left(p(Y|x)||p(Y|c)\right) \qquad \text{(A.5)}
\end{aligned}$$

Consider the IB criterion in Equation (3.1); i.e, the maximization of $I(C,Y) - \frac{1}{\beta} I(X,C)$. This can be rewritten as a minimization in the following form: $\min[I(X,C) - \beta I(C,Y)]$ which is equivalent to $\min[I(X,C) + \beta.\left(I(X,Y) - I(C,Y)\right)]$ since $I(X,Y)$ is a constant for the minimization. Using the result of Equation A.5 the optimization becomes:

$$\min \left[I(X,C) + \beta \sum_{x,c} p(x,c) KL\left(p(Y|x)||p(Y|c)\right) \right]$$

A.3 Re-estimation Formula for HMM/KL Realignment

In this section, we compute the relevance variable distributions in each cluster $p(Y|c_t)$, that is the solution of the minimization of Equation (4.10). The minimization is given by:

$$\min \sum_t p(s_t) KL(p(Y|s_t)||p(Y|c_t)) \tag{A.6}$$

Consider the objective function:

$$\begin{aligned}\sum_t p(s_t) KL(p(Y|s_t)||p(Y|c_t)) &= \sum_i \sum_{\{s_t : s_t \in c_i\}} p(s_t) KL(p(Y|s_t)||p(Y|c_i)) \\ &= \sum_i \sum_{\{s_t : s_t \in c_i\}} p(s_t) \sum_j p(y_j|s_t) \log \left(\frac{p(y_j|s_t)}{p(y_j|c_i)} \right)\end{aligned}$$

Introducing Lagrange multipliers for the constraints $\sum_j p(y_j|c_i) = 1; \forall i$ the objective function becomes:

$$\mathcal{J} = \sum_i \sum_{\{s_t : s_t \in c_i\}} p(s_t) \sum_j p(y_j|s_t) \log \left(\frac{p(y_j|s_t)}{p(y_j|c_i)} \right) + \sum_i \lambda_i \left(\sum_j p(y_j|c_i) - 1 \right) \tag{A.7}$$

Taking partial derivatives with respect to particular variable $p(y_l|c_m)$, i.e., $\frac{\partial \mathcal{J}}{\partial p(y_l|c_m)} = 0$ results in:

$$\begin{aligned}\sum_{\{s_t : s_t \in c_m\}} p(s_t) - \frac{p(y_l|s_t)}{p(y_l|c_m)} + \lambda_m = 0 \\ p(y_l|c_m) = \frac{1}{\lambda_m} \sum_{\{s_t : s_t \in c_m\}} p(s_t) p(y_l|s_t)\end{aligned} \tag{A.8}$$

Substituting the result in the constraint $\sum_l p(y_l|c_m) = 1$

$$\begin{aligned}\sum_l \frac{1}{\lambda_m} \sum_{\{s_t : s_t \in c_m\}} p(s_t) p(y_l|s_t) = 1 \\ \frac{1}{\lambda_m} \sum_{\{s_t : s_t \in c_m\}} p(s_t) \sum_l p(y_l|s_t) = 1 \\ \lambda_m = \sum_{\{s_t : s_t \in c_m\}} p(s_t) = p(c_m)\end{aligned} \tag{A.9}$$

Combining (A.8) and (A.9) we obtain:

$$p(y_l|c_m) = \frac{1}{p(c_m)} \sum_{\{s_t : s_t \in c_m\}} p(s_t) p(y_l|s_t) \tag{A.10}$$

The second derivative $\frac{\partial^2 \mathcal{J}}{\partial p(y_l|c_m)^2}$ turns out to be:

$$\sum_{\{s_t : s_t \in c_m\}} p(s_t) \frac{p(y_l|s_t)}{[p(y_l|c_m)]^2} \tag{A.11}$$

which is positive since all terms in the sum are positive. This confirms that the optimum obtained in Equation (A.10) is a minimum.

Bibliography

[1] Adami, A., Kajarekar, S., and Hermansky, H. (2002). A new speaker change detection method for two speaker segmentation. In *IEEE International Conference on Acoustics Speech and Signal Processing*, volume 4, pages 3908–3911.

[2] Ajmera, J. (2004). *Robust Audio Segmentation*. Ph.D. thesis, Ecole Polytechnique Federale de Lausanne (EPFL).

[3] Ajmera, J. and Wooters, C. (2003). A robust speaker clustering algorithm. In *IEEE Automatic Speech Recognition Understanding (ASRU '03) Workshop, Virgin Islands, USA*, pages 411–416.

[4] Ajmera, J., McCowan, I., and Bourlard, H. (2004). Robust speaker change detection. *Signal Processing Letters, IEEE*, **11**(8), 649–651.

[5] Anguera, X. (2006). *Robust Speaker Diarization for Meetings*. Ph.D. thesis, Universitat Politecnica de Catalunya.

[6] Anguera, X. and Hernando, J. (2004). Evolutive speaker segmentation using a repository system. In *Proc. International Conference on Speech and Language Processing, Jeju Island, Korea*. IEEE.

[7] Anguera, X. and Hernando, J. (2005). XBIC: Real-Time Cross Probabilities Measure for Speaker Segmentation.

[8] Anguera, X., Aguilo, M., Wooters, C., Nadeu, C., and Hernando, J. (2006a). Hybrid speech/nonspeech detector applied to speaker diarization ofmeetings. In *Speaker and Language Recognition Workshop*. IEEE Odyssey.

[9] Anguera, X., Wooters, C., and Hernando, J. (2006b). Purity algorithms for speaker diarization of meetings data.

[10] Anguera, X., Wooters, C., and Pardo, J. (2006c). Robust speaker diarization for meetings: ICSI RT06s evaluation system. In *Ninth International Conference on Spoken Language Processing*.

[11] Anguera, X., Wooters, C., Peskin, B., and Aguiló, M. (2006d). Robust speaker segmentation for meetings: The ICSI-SRI spring 2005 diarization system. *Lecture Notes in Computer Science*, **3869**, 402.

[12] Anguera, X., Wooters, C., and Hernando, J. H. (2006e). Speaker diarization for multi-party meetings using acoustic fusion. In *Proceedings of Automatic Speech Recognition and Understanding*, pages 426–431.

[13] Aradilla, G. (2008). *Acoustic Models for Posterior Features in Speech Recognition*. Ph.D. thesis, Ecole Polytechnique Fédérale de Lausanne, Lausanne , Switzerland.

[14] Athineos, M. and Ellis, D. (2003). Frequency-domain linear prediction for temporal features. In *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU '03*.

[15] Barras, C., Zhu, X., Meignier, S., and Gauvain, J. (2004). Improving speaker diarization. In *RT-04F workshop*.

[16] Barron, A., Rissanen, J., and B., Y. (1998). The minimum description length principle in coding and modeling. *IEEE Transactions on Information Theory*, **44**, 2743–2760.

[17] Ben, M., Betser, M., Bimbot, F., and Gravier, G. (2004). Speaker Diarization using bottom-up clustering based on a Parameter-derived Distance between adapted GMMs. In *Eighth International Conference on Spoken Language Processing*. ISCA.

[18] Bimbot, F. and Mathan, L. (1993). Text-free speaker recognition using an arithmetic-harmonic sphericity measure. In *Third European Conference on Speech Communication and Technology*. ISCA.

[19] Boakye, K., Vinyals, O., and Friedland, G. (2008). Two's a crowd: Improving speaker diarization by automatically identifying and excluding overlapped speech. In *Proceedings of Interspeech 2008, Brisbane Australia*, pages 32–35. ISCA.

[20] Bonastre, J., Delacourt, P., Fredouille, C., Merlin, T., and Wellekens, C. (2000). A speaker tracking system based on speaker turn detection for NIST evaluation. In *Proceedings of the Acoustics, Speech, and Signal Processing, 2000. on IEEE International Conference-Volume 02*. IEEE Computer Society.

[21] Bozonnet, S., Evans, N., Anguera, X., Vinyals, O., and Fredouille, G. F. C. (2010). System Output Combination for Improved Speaker Diarization. In *Proceedings of Interspeech*. ISCA.

[22] Burger, S., MacLaren, V., and Yu, H. (2002). The ISL meeting corpus: The impact of meeting type on speech style. In *Seventh International Conference on Spoken Language Processing*. `http://www.is.cs.cmu.edu/meeting_room/corpus`.

[23] Chen, L., Rose, R., Qiao, Y., Kimbara, I., Parrill, F., Welji, H., Han, T., Tu, J., Huang, Z., Harper, M., *et al.* (2006). VACE multimodal meeting corpus. *Machine Learning for Multimodal Interaction*, pages 40–51.

[24] Chen, S. and Gopalakrishnan, P. (1998). Speaker, environment and channel change detection and clustering via the bayesian information criterion. In *Proceedings of DARPA speech recognition workshop*, pages 127–138.

[25] Cheng, S. and Wang, H. (2004). METRIC-SEQDAC: A hybrid approach for audio segmentation. In *Proc. International Conference on Spoken Language Processing*.

[26] Chu, S., Tang, H., and Huang, T. (2009). Fishervoice and semi-supervised speaker clustering. In *Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4089–4092. IEEE Computer Society.

[27] Cover, T. M. and Thomas, J. A. (1991). *Elements of Information Theory*. John Wiley & sons.

[28] Delacourt, P. and Wellekens, C. (2000). DISTBIC: A speaker-based segmentation for audio data indexing* 1. *Speech communication*, **32**(1-2), 111–126.

[29] Delacourt, P., Kryze, D., and Wellekens, C. (1999). Detection of speaker changes in an audio document. In *Sixth European Conference on Speech Communication and Technology*. ISCA.

[30] Dines, J., Vepa, J., and Hain, T. (2006). The segmentation of multi-channel meeting recordings for automatic speech recognition. In *Ninth International Conference on Spoken Language Processing*. IEEE.

[31] Evans, N., Fredouille, C., and Bonastre, J. (2009). Speaker Diarization Using Unsupervised Discriminant Analysis of Inter-channel Delay Features. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2009. ICASSP 2009, Taipei*, pages 4061 – 4064. IEEE Computer Society.

[32] Fiscus, J., Ajot, J., Michel, M., and Garofolo, J. (2006). The Rich Transcription 2006 Spring Meeting Recognition Evaluation. *Lecture Notes in Computer Science*, **4299**, 309.

[33] Fiscus, J., Ajot, J., and Garofolo, J. (2008). The rich transcription 2007 meeting recognition evaluation. *Multimodal Technologies for Perception of Humans, Lecture Notes in Computer Science, Berlin*.

[34] Friedland, G., Hung, H., and Yeo, C. (2009a). Multi-modal speaker diarization of real-world meetings using compressed-domain video features (PDF).

[35] Friedland, G., Vinyals, O., Huang, Y., and Müller, C. (2009b). Prosodic and other Long-Term Features for Speaker Diarization. *IEEE Transactions on Audio, Speech, and Language Processing*, **17**(5), 985–993.

[36] Gangadharaiah, R., Narayanaswamy, B., and Balakrishnan, N. (2004). A novel method for two-speaker segmentation. In *Eighth International Conference on Spoken Language Processing*.

[37] Garau, G. and Bourlard, H. (2010). Using audio and visual cues for speaker diarisation initialisation. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 4942–4945. IEEE.

[38] Garofolo, J., Laprun, C., Michel, M., Stanford, V., and Tabassi, E. (2004). The NIST meeting room pilot corpus. In *Proc. 4th Intl. Conf. on Language Resources and Evaluation*. `http://www.itl.nist.gov/iad/mig/test_beds/meeting_corpus_1/index.html`.

[39] Gauvain, J., Lamel, L., and Adda, G. (1998). Partitioning and transcription of broadcast news data. In *ICSLP '98*, volume 5, pages 1335–1338.

[40] Gish, H., Siu, M., and Rohlicek, R. (1991). Segregation of speakers for speech recognition and speakeridentification. In *1991 International Conference on Acoustics, Speech, and Signal Processing, 1991. ICASSP-91.*, pages 873–876.

[41] Gish, H., Schmidt, M., and Mielke, A. (1994). A robust, segmental method for text independent speaker identification. In *1994 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1994. ICASSP-94.*

[42] Glenn, M. and Strassel, S. (2009). Shared linguistic resources for the meeting domain. *Multimodal Technologies for Perception of Humans*, pages 401–413.

[43] Goldberger, J., Greenspan, H., and Gordon, S. (2002). Unsupervised image clustering using the information bottleneck method. In *Proceedings of the 24th DAGM Symposium on Pattern Recognition*, pages 158–165.

[44] Gordon, S., Greenspan, H., and Goldberger, J. (2008). Applying the information bottleneck principle to unsupervised clustering of discrete and continuous image representations. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 370–377. IEEE.

[45] Hain T. et. al. (2006). The AMI meeting transcription system: Progress and performance. In *Proceedings of NIST RT'O6 Workshop*.

[46] Harremoës, P. and Tishby, N. (2007). The Information bottleneck revisited or how to choose a good distortion measure. In *IEEE International Symposium on Information Theory, 2007. ISIT 2007*, pages 566–570.

[47] Hecht, R. and Tishby, N. (2005). Extraction of relevant speech features using the information bottleneck method. In *Ninth European Conference on Speech Communication and Technology*.

[48] Heck, L., Sankar, A., and Menlo Park, C. (1997). Acoustic clustering and adaptation for robust speech recognition. In *Fifth European Conference on Speech Communication and Technology*.

[49] Hermansky, H. and Morgan, N. (1994). RASTA processing of speech. *IEEE transactions on speech and audio processing*, **2**(4), 578–589.

[50] Huijbregts, M., Ordelman, R., and de Jong, F. (2007). Annotation of heterogeneous multimedia content using automatic speech recognition. *Semantic Multimedia*, pages 78–90.

[51] Imseng, D. and Friedland, G. (2010). An Adaptive Initialization Method for Speaker Diarization based on Prosodic Features. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4946–4949.

[52] Istrate, D., Fredouille, C., Meignier, S., Besacier, L., and Bonastre, J. (2006). NIST RT05S Evaluation: Pre-processing techniques and speaker diarization on multiple microphone meetings. *Machine Learning for Multimodal Interaction*, pages 428–439.

[53] Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., *et al.* (2003). The ICSI meeting corpus. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03)*. `http://www.icsi.berkeley.edu/speech/mr`.

[54] Jin, H., Kubala, F., and Schwartz, R. (1997). Automatic speaker clustering. In *Proceedings of the DARPA speech recognition workshop*, pages 108–111.

[55] Jin, Q. and Schultz, T. (2004). Speaker segmentation and clustering in meetings. In *Eighth International Conference on Spoken Language Processing*. ISCA.

[56] Johnson, S. (1999). Who spoke when?-automatic segmentation and clustering for determining speaker turns. In *Sixth European Conference on Speech Communication and Technology*.

[57] Johnson, S. and Woodland, P. (1998). Speaker clustering using direct maximisation of the MLLR-adapted likelihood. In *Fifth International Conference on Spoken Language Processing*.

[58] Juang, B. and Rabiner, L. (1985). A probabilistic distance measure for hidden Markov models. *AT&T Bell Laboratories technical journal*, **64**(2), 391–408.

[59] Junqua, J., Mak, B., and Reaves, B. (1994). A robust algorithm for word boundary detection in the presence of noise. *IEEE Transactions on speech and audio processing*, **2**(3), 406–412.

[60] Kim, H., Ertelt, D., and Sikora, T. (2005). Hybrid speaker-based segmentation system using model-level clustering. In *Proc. 2005 IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, volume 1, pages 745–748.

[61] Lamel, L., Rabiner, L., Rosenberg, A., and Wilpon, J. (1981). An improved endpoint detector for isolated word recognition. *IEEE Transactions on Acoustics Speech and Signal Processing*, **29**(4), 777–785.

[62] Lapidot, I. (2003). SOM as likelihood estimator for speaker clustering. In *Eighth European Conference on Speech Communication and Technology*, pages 3001–3004. ISCA.

[63] Laskowski, K. and Schultz, T. (2006). Unsupervised learning of overlapped speech model parameters for multichannel speech activity detection in meetings. In *ICASSP*, pages 993–996.

[64] Laskowski, K., Jin, Q., and Schultz, T. (2004). Crosscorrelation-based multispeaker speech activity detection. In *Eighth International Conference on Spoken Language Processing*.

[65] Lathoud, G. and McCowan, I. (2003). Location based speaker segmentation. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*.

[66] Lin, J. (1991). Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, **37**(1), 145–151.

[67] Lu, L. and Zhang, H. (2002). Real-time unsupervised speaker change detection. *Pattern Recognition*, **2**, 20358.

[68] Malegaonkar, A., Ariyaeeinia, A., Sivakumaran, P., and Fortuna, J. (2006). Unsupervised speaker change detection using probabilistic pattern matching. *IEEE signal processing letters*, **13**(8), 509.

[69] McCowan, I., Carletta, J., Kraaij, W., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., *et al.* (2005). The AMI meeting corpus. In *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*, volume 88. `http://corpus.amiproject.org/documentations/overview/`.

[70] Meignier, S., Bonastre, J., and Igounet, S. (2001). E-HMM approach for learning and adapting sound models for speaker indexing. In *2001: A Speaker Odyssey-The Speaker Recognition Workshop*.

[71] Meignier, S., Moraru, D., Fredouille, C., Bonastre, J.-F., and Besacier, L. (2006). Step-by-step and integrated approaches in broadcast news speaker diarization. *Computer Speech and*

Language, **20**(2-3), 303 – 330. Odyssey 2004: The speaker and Language Recognition Workshop - Odyssey-04.

[72] Misra, H., Bourlard, H., and Tyagi, V. (2003). New entropy based combination rules in HMM/ANN multi-stream ASR. In *Proceedings ICASSP*, volume 3, pages 1–5.

[73] Moh, Y., Nguyen, P., and Junqua, J. (2003). Towards domain independent speaker clustering. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 85–88.

[74] Moraru, D., Meignier, S., Besacier, L., Bonastre, J., and Magrin-Chagnolleau, I. (2003). The ELISA consortium approaches in speaker segmentation during the NIST 2002 speaker recognition evaluation. In *ICASSP'03*.

[75] Moraru, D., Meignier, S., Fredouille, C., Besacier, L., and Bonastre, J. (2004). The ELISA consortium approaches in broadcast news speaker segmentation during the NIST 2003 rich transcription evaluation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04)*, volume 1.

[76] Moraru, D., Ben, M., and Gravier, G. (2005). Experiments on speaker tracking and segmentation in radio broadcast news. In *Ninth European conference on speech communication and technology*. ISCA.

[77] Mostefa, D., Moreau, N., Choukri, K., Potamianos, G., Chu, S., Tyagi, A., Casas, J., Turmo, J., Cristoforetti, L., Tobia, F., *et al.* (2007). The chil audiovisual corpus for lecture and meeting analysis inside smart rooms. *Language Resources and Evaluation*, **41**(3), 389–407.

[78] Ng, A., Jordan, M., and Weiss, Y. (2001). On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14: Proceeding of the 2001 Conference*, pages 849–856.

[79] Nguyen, P., Rigazio, L., Moh, Y., and Junqua, J. (2002). Rich Transcription 2002 Site Report, Panasonic Speech Technology Laboratory (PSTL). In *Proc. 2002 Rich Transcription Workshop (RT-02)*.

[80] Ning, H., Liu, M., Tang, H., and Huang, T. (2006a). A Spectral Clustering Approach to Speaker Diarization. In *Ninth International Conference on Spoken Language Processing*, pages 2178–2181. ISCA.

[81] Ning, H., Xu, W., Gong, Y., and Huang, T. (2006b). Improving speaker diarization by cross EM refinement. In *2006 IEEE International Conference on Multimedia and Expo*, pages 1901–1904. IEEE.

[82] Ning, H., Xu, W., Gong, Y., and Huang, T. (2006c). Improving Speaker Diarization by Cross EM Refinement. In *IEEE International Conference on Multimedia and Expo*, pages 1901–1904.

[83] Nishida, M. and Kawahara, T. (2003). Unsupervised speaker indexing using speaker model selection based on Bayesian information criterion. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 172–175. IEEE.

[84] NIST (2004). Spring 2004 (RT-04S) Rich Transcription Meeting Recognition Evaluation Plan. http://www.itl.nist.gov/iad/mig//tests/rt/2004-spring/documents/rt04s-meeting-eval-plan-v1.pdf.

[85] NIST (2006). Spring 2006 (RT-06S) Rich Transcription Meeting Recognition Evaluation Plan. http://www.itl.nist.gov/iad/mig//tests/rt/2006-spring/docs/rt06s-meeting-eval-plan-V2.pdf.

[86] Otterson, S. (2007). Improved location features for meeting speaker diarization. In *Proceedings of INTERSPEECH*, pages 1849–1852.

[87] Otterson, S. and Ostendorf, M. (2007). Efficient use of overlap information in speaker diarization. In *IEEE Workshop on Automatic Speech Recognition & Understanding, 2007. ASRU*, pages 683–686. IEEE Computer Society.

[88] Pardo, J., Anguera, X., and Wooters, C. (2006a). Speaker diarization for multi-microphone meetings: Mixing acoustic features and inter-channel time differences. In *International Conference on Speech and Language Processing*.

[89] Pardo, J., Anguera, X., and Wooters, C. (2006b). Speaker diarization for multi-microphone meetings using only between-channel differences. In *MLMI*.

[90] Pardo, J., Anguera, X., and Wooters, C. (2007). Speaker Diarization For Multiple-Distant-Microphone Meetings Using Several Sources of Information. *IEEE Transactions on Computers*, **56**(9), 1212–1224.

[91] Pelecanos, J. and Sridharan, S. (2001). Feature warping for robust speaker verification. In *2001: A Speaker Odyssey-The Speaker Recognition Workshop*.

[92] Pfau, T., Ellis, D., and Stolcke, A. (2001). Multispeaker speech activity detection for the ICSI meeting recorder. In *Proceedings of ASRU*, volume 1.

[93] Reynolds, D. (1997). Comparison of background normalization methods for text-independent speaker verification. In *Fifth European Conference on Speech Communication and Technology*. ISCA.

[94] Reynolds, D., Quatieri, T., and Dunn, R. (2000). Speaker Verification Using Adapted Gaussian Mixture Models. *Digital Signal Processing*, **10**(1-3), 19–41.

[95] Roch, M. and Cheng, Y. (2004). Speaker segmentation using the MAP-adapted bayesian information criterion. In *ODYSSEY04-The Speaker and Language Recognition Workshop*. ISCA.

[96] Rougui, J., Rziza, M., Aboutajdine, D., Gelgon, M., and Martinez, J. (2006). Fast incremental clustering of gaussian mixture speaker models for scaling up retrieval in on-line broadcast.

[97] Samuel, T., Sriram, G., and Hynek, H. (2009). Tandem representations of spectral envelope and modulation frequency features for asr. In *Proceedings of Interspeech, Brighton, UK*.

[98] Sankar, A., Beaufays, F., and Digalakis, V. (1995). Training data clustering for improved speech recognition. In *Fourth European Conference on Speech Communication and Technology*.

[99] Sankar, A., Weng, F., Rivlin, Z., Stolcke, A., and Gadde, R. (1998). The development of SRI's 1997 Broadcast News transcription system. In *Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, pages 91–96.

[100] Schneidman, E., Slonim, N., Tishby, N., van Steveninck, R., and Bialek, W. (2001). Analyzing neural codes using the information bottleneck method.

[101] Schwartz, G. (1978). Estimation of the dimension of a model. *Annals of Statistics*, **6**, 461–464.

[102] Seldin, Y., Slonim, N., and Tishby, N. (2007). Information bottleneck for non co-occurrence data. In *Advances in Neural Information Processing Systems 19*. MIT Press.

[103] Siegler, M., Jain, U., Raj, B., and Stern, R. (1997). Automatic segmentation, classification and clustering of broadcast news audio. In *Proc. DARPA speech recognition workshop*, volume 1997.

[104] Sinha, R., Tranter, S., Gales, M., and Woodland, P. (2005). The cambridge university march 2005 speaker diarisation system. In *Ninth European Conference on Speech Communication and Technology*.

[105] Sivakumaran, P., Fortuna, J., and Ariyaeeinia, A. (2001). On the use of the bayesian information criterion in multiple speaker detection. In *Seventh European Conference on Speech Communication and Technology*.

[106] Slonim, N. (2002). *The Information Bottleneck: Theory and Applications*. Ph.D. thesis, The Hebrew University of Jerusalem.

[107] Slonim, N. and Tishby, N. (2000). Document clustering using word clusters via the information bottleneck method. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 208–215. ACM.

[108] Slonim, N. and Tishby, N. (2001). The power of word clusters for text classification. In *Proceedings of ECIR-01, 23rd European Colloquium on Information Retrieval Research*.

[109] Slonim, N., Friedman, N., and Tishby, N. (1999). Agglomerative information bottleneck. In *Proceedings of Advances in Neural Information Processing Systems*, pages 617–623. MIT Press.

[110] Slonim, N., F., F., and N., T. (2002a). Unsupervised document classification using sequential information maximization. In *Proceeding of SIGIR'02, 25th ACM intermational Conference on Research and Development of Information Retireval*.

[111] Slonim, N., Friedman, N., and Tishby, N. (2002b). Unsupervised document classification using sequential information maximization. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 129–136. ACM.

[112] Solomonoff, A., Mielke, A., Schmidt, M., and Gish, H. (1998). Clustering speakers by their voices. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, 1998*, volume 2.

[113] Sriram, G., Samuel, T., and Hynek, H. (2008). Front-end for far-field speech recognition based on frequency domain linear prediction. In *Proceedings of INTERSPEECH, Brisbane, Australia*.

[114] Sun, H., Nwe, T., Ma, B., and Li, H. (2009). Speaker diarization for meeting room audio. In *International conference on Speech and Language Processing*, pages 900–903.

[115] Sun, H., Ma, B., Khine, S., and Li, H. (2010). Speaker Diarization System for RT07 and RT09 Meeting Room Audio.

[116] Thomas, S., Ganapathy, S., and Hermansky, H. (2008). Recognition of Reverberant Speech Using Frequency Domain Linear Prediction. *IEEE Signal Processing Letters*, **15**, 681–684.

[117] Tishby, N., Pereira, F., and Bialek, W. (1998). The information bottleneck method. In *NEC Research Institute TR*.

[118] Tranter, S. (2005). Two-way cluster voting to improve speaker diarisation performance. In *Proc. ICASSP*, pages 753–756.

[119] Tranter, S. and Reynolds, D. (2006). An overview of automatic speaker diarisation systems. *IEEE Transactions on Audio, Speech and Language Processing*, **14**, 1557–1565.

[120] Tranter, S. and Reynolds, D. (2004). Speaker diarization for broadcast news. In *ODYSSEY '04, Toledo, Spain*.

[121] Tritschler, A. and Gopinath, R. (1999). Improved speaker segmentation and segments clustering using the bayesian information criterion. In *Sixth European Conference on Speech Communication and Technology*.

[122] Tsai, W. and Wang, H. (2006). On maximizing the within-cluster homogeneity of speaker voice characteristics for speech utterance clustering. In *2006 IEEE International Conference on Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings*, volume 1.

[123] Valente, F. (2006). Infinite models for speaker clustering. In *Ninth International Conference on Spoken Language Processing*.

[124] Valente, F. and Wellekens, C. (2004). Variational bayesian speaker clustering. In *ODYSSEY04-The Speaker and Language Recognition Workshop*. ISCA.

[125] Valente, F. and Wellekens, C. (2005). Variational bayesian adaptation for speaker clustering. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, Lisbon, Portugal*.

[126] Valente, F., Motlicek, P., and Vijayasenan, D. (2010). Variational Bayesian Speaker Diarization of Meeting Recordings. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, Dallas, Texas, USA*.

[127] van Leeuwen, D. (2006). The TNO speaker diarization system for NIST RT05s meeting data. *Machine Learning for Multimodal Interaction*, pages 440–449.

[128] van Leeuwen, D. and Huijbregts, M. (2006). The AMI speaker diarization system for NIST RT06s meeting data. *Lecture Notes in Computer Science*, **4299**, 371.

[129] van Leeuwen, D. and Konecnỳ, M. (2008). Progress in the AMIDA speaker diarization system for meeting data. *Multimodal Technologies for Perception of Humans: International Evaluation Workshops Clear 2007 and Rt 2007, Baltimore, MD, USA, May 8-11, 2007, Revised Selected Papers*, page 475.

[130] Vandecatseye, A. and Martens, J. (2003). A fast, accurate and stream-based speaker segmentation and clustering algorithm. In *Eighth European Conference on Speech Communication and Technology*.

[131] Vijayasenan, D., Valente, F., and Bourlard, H. (2007). Agglomerative information bottleneck for speaker diarization of meetings data. In *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 250–255.

[132] Vijayasenan, D., Valente, F., and Bourlard, H. (2008a). Combination of agglomerative and sequential clustering for speaker diarization. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4361–4364.

[133] Vijayasenan, D., Valente, F., and Bourlard, H. (2008b). Integration of tdoa features in information bottleneck framework for fast speaker diarization. In *Interspeech 2008*.

[134] Vijayasenan, D., Valente, F., and Bourlard, H. (2009a). An information theoretic approach to speaker diarization of meeting data. *IEEE Transactions on Audio, Speech and Language Processing*, **17**(7), 1382 – 1393.

[135] Vijayasenan, D., Valente, F., and Bourlard, H. (2009b). KL realignment for speaker diarization with multiple feature streams. In *10th Annual Conference of the International Speech Communication Association*.

[136] Vijayasenan, D., Valente, F., and Bourlard, H. (2010a). Advances in fast multistream diarization based on the information bottleneck framework. In *International Conference on Spoken Language Processing (INTERSPEECH)*. ISCA.

[137] Vijayasenan, D., Valente, F., and Bourlard, H. (2010b). Multistream Speaker Diarization beyond Two Acoustic Feature Streams. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE.

[138] Vijayasenan, D., Valente, F., and Bourlard, H. (2011). An information theoretic combination of mfcc and tdoa features for speaker diarization. To Appear*IEEE Transactions on Audio, Speech and Language Processing*, **19**(2), 431–438. `http://dx.doi.org/10.1109/TASL.2010.2048603`.

[139] Vinyals, O. and Friedland, G. (2008). Modulation spectrogram features for speaker diarization. In *Proceedings of Interspeech*.

[140] Wilcox, L., Chen, F., Kimber, D., and Balasubramanian, V. (1994). Segmentation of speech using speaker identification. In *1994 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1994. ICASSP-94.*, volume 1.

[141] Willsky, A. and Jones, H. (1976). A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. *IEEE Transactions on Automatic Control*, **21**, 108–112.

[142] Wooters, C. and Huijbregts, M. (2008). The ICSI RT07s speaker diarization system. *Lecture Notes in Computer Science*, **4625**, 509–519.

[143] Wooters, C. and Huijbregts, M. (2009). The ICSI RT07s speaker diarization system. *Multimodal Technologies for Perception of Humans*, pages 509–519.

[144] Wooters, C., Fung, J., Peskin, B., and Anguera, X. (2004). Towards robust speaker segmentation: The ICSI-SRI fall 2004 diarization system. In *RT-04F Workshop*, volume 23.

[145] Wrigley, S. N., Brown, G. J., Wan, V., and Renals, S. (2005). Speech and crosstalk detection in multichannel audio. *IEEE Transactions on speech and audio processing*, **13**(1), 84–91.

[146] X., A. and Wooters, C.and Hernando, J. (2006). Purity algorithms for speaker diarization of meetings data. In *Proceedings of ICASSP*.

[147] X. Anguera (2006). Beamformit, the fast and robust acoustic beamformer. In *http://www.icsi.berkeley.edu/x̃anguera/BeamformIt*.

[148] Yamaguchi, M., Yamashita, M., and Matsunaga, S. (2005). Spectral cross-correlation features for audio indexing of broadcast news and meetings. In *International conference on Speech and Language Processing*. ISCA.

[149] Zhou, B. and Hansen, J. (2000). Unsupervised audio stream segmentation and clustering via the Bayesian information criterion. In *Sixth International Conference on Spoken Language Processing*.

[150] Zhou, B. and Hansen, J. (2005). Efficient audio stream segmentation via the combined T 2 statistic and Bayesian information criterion. *IEEE Transactions on Speech and Audio Processing*, **13**(4), 467.

[151] Zhu, X., Barras, C., Meignier, S., and Gauvain, J. (2005). Combining speaker identification and BIC for speaker diarization. In *Ninth European Conference on Speech Communication and Technology*.

[152] Zhu, X., Barras, C., Lamel, L., and Gauvain, J. (2006). Speaker diarization: From broadcast news to lectures. *Machine Learning for Multimodal Interaction*, pages 396–406.

[153] Zochová, P. and Radová, V. (2005). Modified DISTBIC algorithm for speaker change detection. In *International Conference on Spoken Language Processing*. ISCA.

Curriculum Vitae

Deepu Vijayasenan

Idiap Research Institute
PO Box 592 CH - 1920
Martigny, Switzerland
Ph: +41 786 15 77 09
dvijaya@idiap.ch

Education

2006 - **Ph.D** Student,
Ecole Polytechnique Fédéral de Lausanne (EPFL), Switzerland
(at Idiap Research Institute expected to complete by Nov 2010)

2001 - '03 **M.E**, Signal Processing, Indian Institute of Science, Bangalore
Project: Online Handwritten Character Recognition for Tamil

1996 -2000 **B.Tech**, Electronics and Communication, University of Kerala,

Professional Experience

Feb 2003 - Nov 2006

Consultant - Hewlett-Packard Research Laboratories (HP Labs), Bangalore

Jun 2000 - July 2001

Engineer - VLSI, WIPRO Technologies Pvt. Ltd., Bangalore

List of Publications

Journals Publications

- Deepu Vijayasenan F. Valente and Hervé Bourlard "An Information Theoretic Approach to Speaker Diarization of Meeting Data" 17(7) IEEE Transactions on Acoustics, Speech and Language Processing, September 2009 pp. 1382 - 1393, 10.1109/TASL.2009.2015698
- Deepu Vijayasenan F. Valente and Hervé Bourlard "An Information Theoretic Combination of MFCC and TDOA Features for Speaker Diarization" To appear in IEEE Transactions on Acoustics, Speech and Language Processing, 10.1109/TASL.2010.2048603

Conference Publications

- Deepu Vijayasenan F. Valente and Hervé Bourlard "Advances in Fast Multistream Diarization based on the Information Bottleneck Framework" to appear in Proceedings of Interpseech 2010
- Deepu Vijayasenan F. Valente and Hervé Bourlard "Multistream Speaker Diarization Beyond Two Acoustic Streams" Proceedings of ICASSP 2010
- Deepu Vijayasenan F. Valente and Hervé Bourlard "KL Realignment for Speaker Diarization with Multiple Feature Streams" Proceedings of Interspeech 2009 pp. 1059 - 1062
- Deepu Vijayasenan F. Valente and Hervé Bourlard "Mutual Information based Channel Selection for Speaker Diarization of Meetings Data" ICASSP 2009 pp.4065-4068
- Deepu Vijayasenan F. Valente and Hervé Bourlard "Integration of TDOA Features in Information Bottleneck Framework for Fast Speaker Diarization" in Proceedings of Interspeech, 2008 pp. 40-43
- Deepu Vijayasenan F. Valente and Hervé Bourlard "Combination of Agglomerative and Sequential Clustering for Speaker Diarization", in Proceedings of ICASSP, 2008, pp. 4361-4364

- Deepu Vijayasenan F. Valente and Hervé Bourlard “Agglomerative Information Bottleneck for Speaker Diarization of Meetings Data”, in Proceedings of ASRU, 2007, pp. 250-255