Speaker diarization of news broacasts and meeting recordings

Koh, Eugene Chin Wei

2009

Koh, E. C. W. (2009). Speaker diarization of news broacasts and meeting recordings. Master’ s thesis, Nanyang Technological University, Singapore.

https://hdl.handle.net/10356/15707

https://doi.org/10.32657/10356/15707

SPEAKER DIARIZATION OF NEWS BROADCASTS AND MEETING RECORDINGS

KOH CHIN WEI, EUGENE

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Master of Engineering

2009

Abstract

Given a piece of audio recording, the task of speaker diarization can be summarized as answering the question of “Who spoke when ?”. This thesis offers a review of the techniques and issues relating to performing speaker diarization on broadcast news recordings, as well as meeting recordings.

The broadcast news domain is generally regarded to be simpler because the turn taking between speakers is better controlled and audio quality tends to be higher. The typical approach used for this domain consist of two steps - speaker segmentation and then speaker clustering. The Bayesian Information Criterion (BIC) has been a very popular distance measure for both speaker segmentation and clustering. Experiments were conducted that confirmed the effectiveness of this distance measure for segmentation and clustering. Further speaker segmentation experiments were performed using the Hotelling's T^2 statistic to augment the BIC. It was observed that while this does speed up processing, the segmentation $FScore$ obtained does not match up to that reported in the literature. A novel speaker clustering approach was also introduced where polynomial expanded feature vectors were used to compute the distance between clusters. It was found that this approach could produce results comparable to that for the BIC.

In order to address the problem of speaker diarization for the meeting domain, a diarization system was developed and submitted for the NIST Rich Transcription 2007 (RT-07) evaluation. This diarization system exploited the diversity of meeting recording channels by performing Time Delay of Arrival (TDOA) estimation using a Normalized Least Means Squared (NLMS) filter. Subsequent performance enhancements were delivered by adding a cluster purification module, as well as a Non-Speech & Silence Removal (NS&SR) module. An overall Diarization Error Rate (DER) of 15.32% was obtained for the RT-07 corpus. This score was found to be competitive against the other entrants in the evaluation exercise.

Acknowledgments

This thesis would not have been possible if not for the kind support, advice and encouragement of the people around me. I would thus like to extend my deepest appreciation to the following individuals.

- My thesis supervisors Asst. Prof. Chng Eng Siong and Dr. Li Haizhou. I have learnt much from their guidance and it is with their support that this thesis has gained fruition.
- Dr. Sun Hanwu & Dr. Nwe Tin Lay have been instrumental to the work done in Chapter 4. Dr. Sun was responsible for developing the bootstrap clustering & cluster purification modules, while Dr. Nwe was responsible for the non-speech and silence removal modules.
- Dr. Tan Yeow Kee who was most supportive and understanding of my need to work on this thesis. It was he who made juggling employment and the thesis possible.

Contents

List of Figures

List of Tables

List of Abbreviations

ACP	Average Cluster Purity
ASC	Average Speaker Coverage
BIC	Bayesian Information Criterion
CMS	Ceptral Mean Subtraction
DER	Diarization Error Rate
DOA	Direction of Arrival
EM	Expectation Maximization
FA	False Alarm speaker time
FAR	False Alarm Rate
GCC-PHAT	Generalized Cross-Correlation using Phase Transform
GLR	Generalized Likelihood Ratio
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
KL distance	Kullback-Leibler distance
LLR	Log-Likelihood Ratio
LPCC	Linear Prediction Cepstral Coefficients
LSP	Line Spectral Pairs
MAP	Maximum a Posteriori
MDM	Multiple Distant Microphones
MDR	Missed Detection Rate
MFCC	Mel-Filterbank Cepstral Coefficient
MS	Missed Speaker time
NIST	National Institute of Standards and Technology
NLMS	Normalized Least Means Squared
NS&SR	Non-Speech & Silence Removal
RT	Rich Transcription
SAD	Speech Activity Detection
SE	Speaker Error time
SNR	Signal to Noise Ratio
SRP-PHAT	Steered Response Power using Phase Transform
TDOA	Time Delay of Arrival
UBM	Universal Background Model
VQ	Vector Quantization

List of Notations

$*$	Convolution
x^*	Complex conjugate of x
$\|x\|$	Modulus
$\mathbf{v}$	Vector variable
$[a_1 \dots a_z]$	A vector consisting of elements a_1 to a_z
$\mathbf{M}$	Matrix variable
$\mathbf{M}[t, i, j]$	Matrix variable with three dimensions t, i & j
$(\cdot)^T$	Matrix or vector transpose
$\mathbf{s}_i[t]$	Discrete time signal i
$s_i[t]$	Sample value of signal $\mathbf{s}_i[t]$ at discrete time t
$\mathbf{s}_j(t)$	Continuous time signal j
Θ_x	A statistical model of the object x
$\mathcal{A}$	A set of objects
$\mathbb{R}^d$	d real number dimensions
$\bar{x}$	The statistical mean of x
$\hat{x}$	An estimate of variable x
$\mathbf{O}_k$	A feature matrix
$\mathbf{o}_{k,i}$	ith feature vector of $\mathbf{O}_k$
$o_{k,i,j}$	j^{th} element of the i^{th} feature vector of $\mathbf{O}_k$

Chapter 1

Introduction

Ever since the invention of the first recording device, humans have been recording their voices for posterity. There thus currently exists a large amount of recorded speech in audio archives around the world. With this large amount of recorded speech, there will thus be the need to better index this speech for retrieval and subsequent processing. Speaker diarization partially fulfills this role by indexing the speech in recordings according to the identities of the speakers present. It essentially answers the question of "Who spoke when ?". Using the time transcriptions of speaker identities, further higher level processing can then be performed. A user can use the speaker transcription to browse an archive according to the identity of interest. This was done in [Kimber *et al.*, 1995] where a speaker-based graphical browsing interface was developed for the navigation of audio recordings. The diarization of speaker identities in a recording can also be used to produce better speech recognition results. This was done for the National Gallery of the Spoken Word project [Hansen *et al.*, 2001] where speaker adaptation was performed on the acoustic models used in speech recognition. A word transcription of the audio archive can consequently be created and navigation of the archive can be done by searching for word phrases.

In the speaker diarization of any recording, good diarization quality can be characterized by having

- Accurate speaker start and stopping time stamps
- Accurate speaker identities
- Accurate identification of non-speech events

In the subsequent chapters, various performance measures will be introduced to quantify these qualities.

Various papers have described speaker diarization techniques for different classes of speech recordings. In [Gish *et al.*, 1991], speaker segmentation and clustering was performed on audio recordings of the radio dialogs between airport traffic controllers and pilots. Papers such as [Ore *et al.*, 2006; Deng *et al.*, 2006] have addressed performing diarization of mixed-channel telephone conversations as a first step to speaker verification in the NIST 2006 Speaker Recognition evaluations [NIST, 2006b]. Two of the most commonly researched domains is that of speaker diarization of broadcast news [Kubala *et al.*, 1998; Cook & Robinson, 1998; Pallett *et al.*, 1998] and meeting recordings [Fiscus *et al.*, 2005; Anguera *et al.*, 2005b; Hain *et al.*, 2005]. This thesis will focus on performing diarization on the domains of broadcast news and meeting recordings.

1.1 Speaker Diarization of Broadcast News and Meetings Recordings

There are four key differences in the nature of the audio recordings for broadcast news and meetings. The techniques used to perform diarization for the two recordings domains will consequently be different as a result of these differences.

- The recording quality for meetings is generally poorer than the same for news broadcasts.

  The audio recordings used for meetings usually are recorded using distant microphones while news broadcasts are recorded off-the-air. The signal-to-noise ratio

(SNR) of meeting recordings are thus typically poorer. The average SNR for meeting recordings of the RT-06s evaluation was estimated in [Anguera, 2006b] to be about 20.77 dB. The same for news broadcasts from the NIST Rich Transcription 2004 Fall (RT-04f) evaluation [NIST, 2004] were estimated to be about 24.22 dB.

- There are potentially more speakers speaking concurrently in a meeting than in a news broadcast.

  The nature of meeting recordings is such that there will be many speakers actively contributing to a meeting. In the RT-06s and RT-07 recordings, there are always four or more participants speaking in any meeting. Meeting participants can sometimes also be heard debating over issues. The quick conversational exchange between participants creates a lot more overlapping speech between speakers. News broadcasts on the other hand typically have only one or two broadcasters actively speaking within each time region. In the event where there are multiple broadcasters, the broadcasters will usually speak in turns with little or no overlap between the speech of different broadcasters. The detection and handling of overlapping speech is thus a much lesser concern for broadcast news.

- Meeting recordings typically contain longer periods of silence than a news broadcast.

  The conversational nature of meetings usually result in it having longer periods of silence than the same for news broadcasts. These extended periods of silence can occur when no participant is actively speaking. The same however is not true for news broadcasts because there generally will always be a broadcaster busy speaking. The extended silences present in meeting recordings thus makes it necessary that a silence detection module be used to improve diarization performance.

- Meeting recordings can have multiple recording channels corresponding to multiple microphones while news broadcasts can only have one.

  The proceedings of a single meeting can be recorded using multiple recording channels. In the context of the NIST RT-06s and RT-07 evaluations, each recording channel will correspond to a single distant microphone placed in a different location in the meeting room. This diversity of recordings thus creates opportunities for algorithms that exploit the differences between channels.

1.2 The Goals of this Thesis

The goals of this thesis are to:

- Review literature about speaker diarization, and the steps of speaker segmentation and speaker clustering.

- Repeat experiments representing the state-of-the-art for speaker segmentation in the broadcast news domain.

- Investigate ways of improving the state-of-the-art for speaker diarization in the meeting room domain.

As listed above, research is concentrated on processing audio from the broadcast news and meeting room domains. The different nature of these two audio domains necessitates the usage of different automatic algorithms.

Research into speaker diarization for the broadcast news domain will be focused on the speaker segmentation step. The first objective for the broadcast news domain would be to replicate the results obtained using the Bayesian Information Criterion (BIC) in [Ajmera *et al.*, 2004]. Using this BIC system as a base, the second objective is then to repeat the results reported in [Zhou & Hansen, 2005] using a hybrid $T^2 + BIC$ system.

The meeting room domain is then explored using recordings from multiple distant microphones. A speaker diarization system will be proposed and the key features of this system would be that:

- It uses the Time Delay of Arrival (TDOA) between channels as a feature for speaker segmentation and clustering.
- Acoustic features are used to refine the speaker segmentation and clustering decisions made using TDOA.

This system will be evaluated in the context of the NIST RT-06s & RT-07 benchmarking efforts. The objective of the experiments will be to obtain a system that can match the overall Diarization Error Rates (DER) reported by other state-of-the-art systems.

1.3 Speaker Segmentation and Clustering

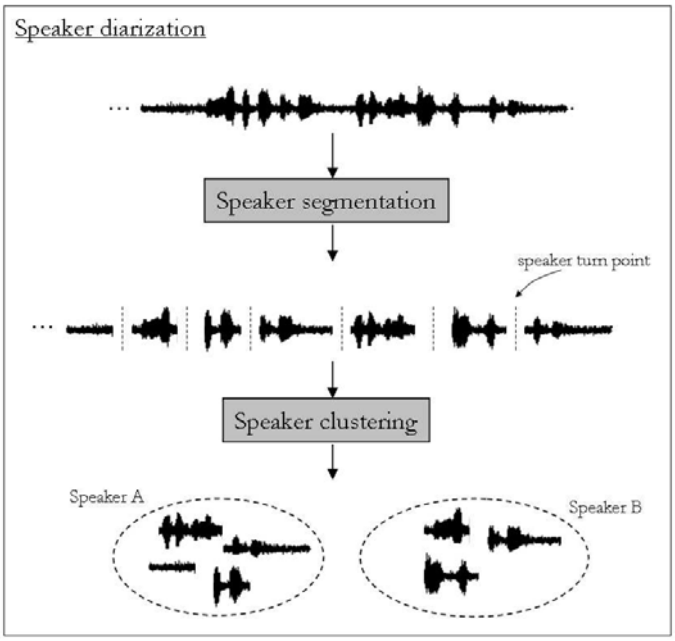


Figure 1.1: Speaker diarization consists of first performing speaker segmentation, then speaker clustering.

The first step in speaker diarization involves breaking up the continuous audio recording into homogenous segments where only a single speaker is speaking. This is typically done by finding locations in the audio recording where there is a speaker transition and a speaker segmentation algorithm will be used to accomplish this automatically. Although the goal in speaker segmentation is to find speaker transitions, segmentation algorithms will usually also detect other acoustic changes such as a change in sampling rate or introduction of background music.

Given a collection of segments resulting from the speaker segmentation process, the next step in speaker diarization would involve clustering the speech segments into clusters of common speakers. The objective when performing speaker clustering would be to produce one, and only one, cluster for each speaker identity. The resultant clusters should ideally be homogenous, i.e. not contain speech segments originating from other speakers.

Chapter 2 examines well known segmentation and clustering techniques reported in the literature for the diarization task. E.g., the Bayesian Information Criterion (BIC)[Chen & Gopalakrishnan, 1998b; Tritschler & Gopinath, 1999; Kemp *et al.*, 2000] and Hotelling's T^2 [Zhou & Hansen, 2000; Zhou & Hansen, 2005; Huang & Hansen, 2004; Huang & Hansen, 2006] approach for segmentation will be examined. The BIC is probably the most commonly cited distance measure for speaker segmentation and clustering while the T^2 has been reported in [Zhou & Hansen, 2000; Huang & Hansen, 2004] to yield good results when combined with the BIC.

1.4 Organization of this Thesis

In Chapter 2, the process of speaker segmentation and clustering will be elaborated and objective measures of segmentation and clustering quality will be introduced. The current techniques for speaker segmentation and clustering will also be reviewed. Chapter 3 then reports on our experiments done using the Hub4-97 database for the broadcast news speaker diarization task. Chapter 4 introduces a speaker diarization system for meeting recordings that was uses the diversity of recording channels to localize the direction of arrival speech. The performance of the system is quantified by performing experiments upon the RT-06s & RT-07 corpora. Conclusions and future works are then summarized in Chapter 5.

Chapter 2

Segmentation and Clustering Techniques for Speaker Diarization

This chapter reviews published techniques for segmentation and clustering used for the speaker diarization task. The structure of this chapter will be as follows. Section 2.1 introduces the segmentation task with Section 2.1.1 discussing common measures used to evaluate the segmentation quality of an algorithm. Section 2.2 then introduces the task of speaker clustering along with the commonly used measures of clustering quality. Sections 2.3, 2.4 and 2.5 will then give an overview of some commonly used speaker segmentation and clustering approaches.

2.1 Speaker Segmentation

Speaker segmentation is the task of breaking up a continuous body of speech along the lines of speaker transitions. This is often useful because it allows the various resulting segments to be processed separately depending on the identity of the speaker present. The technology of speaker segmentation has found widespread usage as an early processing step in many applications. Examples of applications utilizing speaker segmentation are broadcast news archival and indexing [Nishida & Ariki, 1999; Kemp *et al.*, 2000; Wu *et al.*, 2003], speech recognition [Gish *et al.*, 1991; Waheed *et al.*, 2002] and meeting diarizations [Fiscus *et al.*, 2005; Fiscus *et al.*, 2006].

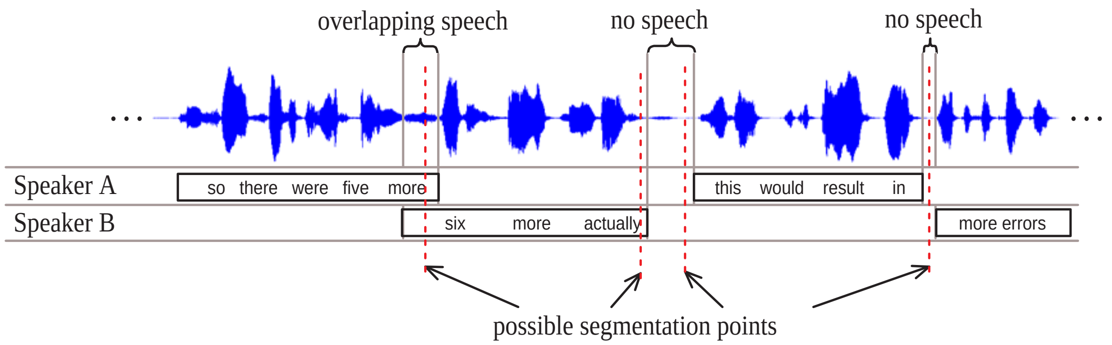


Figure 2.1: The speaker segmentation task. Transitions between speakers could be separated by silence, or be overlapping between speakers.

In [Viswanathan *et al.*, 1999], continuous broadcast news is segmented along speaker transitions and the identity of the speaker in each segment is then identified. This thus allows for convenient indexing of the news corpus according to the broadcaster who reported the news. Subsequent retrieval of the news segment can be done using the identity of the broadcaster. Speaker segmentation also plays a role in speech recognition systems developed in [Meinedo & Neto, 2003b]. In that system, continuous audio is first segmented and the person speaking in each resultant segment is then identified. Speaker specific acoustic models are thereafter used to perform speech recognition. The resultant recognition accuracy improves as a result of better match between the spoken audio and the acoustic model used.

2.1.1 Evaluation measures for speaker segmentation

The measures used to quantify segmentation quality generally are concerned about whether these audio transition points are correctly identified. One commonly used reporting measure would be the False Alarm Rate (FAR) and the Missed Detection Rate (MDR). The FAR and MDR are defined in Eqn. 2.1 and Eqn. 2.2.

$$FAR = \frac{\#\text{ of incorrectly identified turn points}}{\text{total }\#\text{ of turn points identified}} \qquad \text{(Eqn. 2.1)}$$

$$MDR = 1 - \frac{\text{\# of correctly identified turn points}}{\text{total \# of true turn points}} \qquad \text{(Eqn. 2.2)}$$

Papers such as [Chen & Gopalakrishnan, 1998b; Siu *et al.*, 1992; Couvreur & Boite, 1999; Liu & Kubala, 1999] have previously reported their segmentation performance in terms of FAR and MDR. The FAR and MDR values are always bounded between 0 and 1, i.e., a perfect system will yield results of $FAR = 0$ and $MDR = 0$.

Another set of commonly used reporting measure would be the $Precision$, $Recall$ and $FScore$. These measures have been used in such papers as [Kemp *et al.*, 2000; Nishida & Kawahara, 2003; Vandecatseye & Martens, 2003; Ajmera *et al.*, 2004]. The $Precision$ is the fraction of turn points correctly identified, out of all the hypothesized turn points. The $Recall$ is the fraction of turn points correctly identified over the total number of ground-truth turn points. It is thus a measure of the system's ability to detect or "recall" the turn points present. The $Precision$ measure can be obtained from the FAR, as is the $Recall$ from the MDR. The advantage of reporting using these measures is that the $FScore$ serves as a single value summary of both the false positive and the false negative error rates. It thus allows for convenient performance comparisons between different systems.

$$\begin{aligned} Precision &= \frac{\text{\# of correctly identified turn points}}{\text{total \# of turn points identified}} && \text{(Eqn. 2.3)} \\ &= 1 - FAR && \text{(Eqn. 2.4)} \end{aligned}$$

$$\begin{aligned} Recall &= \frac{\text{\# of correctly identified turn points}}{\text{total \# of true turn points}} && \text{(Eqn. 2.5)} \\ &= 1 - MDR && \text{(Eqn. 2.6)} \end{aligned}$$

$$FScore = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \qquad \text{(Eqn. 2.7)}$$

Like the MDR and FAR, the values for $Precision$, $Recall$ and $FScore$ are also bounded between 0 and 1. A perfect system will report values of $Precision = 1$, $Recall = 1$ and consequently, $FScore = 1$. There is usually a trade-off between the $Precision$ and $Recall$ of a system. A system can achieve a high $Recall$ by sacrificing its detection $Precision$, resulting a larger number of turn points that are indeed not true turns (i.e. higher FAR). High $Precision$ can in turn also be achieved by being very conservative about selecting turn points, thus having a low $Recall$ and high MDR. Ideal systems are thus those that can yield a high $Precision$, while having to sacrifice little or no $Recall$.

2.1.2 Factors influencing segmentation performance scoring

2.1.2.1 The evaluation corpus used

In the literature, segmentation results are reported on a multitude of different corpora. [Nishida & Kawahara, 2003] for example reported segmentation $Recall$ and $Precision$ scores of of 0.98 and 0.91 respectively, yielding a $FScore$ of 0.94. The database used in their experiment consisted of 10 hours of Japanese language current affairs discussions. A different database consisting of 1 hour of German news recordings was used in [Kemp *et al.*, 2000]. In that paper, the best $FScore$ achieved was 0.78 using a hybrid segmentation strategy. The corresponding $Recall$ and $Precision$ was 0.67 and 0.93 respectively. While the difference in performance could be attributed to the merits of the algorithm, one must be mindful that the scores were obtained on different corpora and thus it would not be possible to directly compare both systems.

A brief listing of some commonly used standardized corpora that have been used for segmentation reporting can be found in Table 2.1. The variety in reporting corpora thus makes it hard to compare between the performance of different systems. Therefore for the sake of having a fair performance comparison, the Hub4-97 Evaluation [Graff *et al.*, 2002] (Hub4-97e) was selected for the experiments to be reported in our work in

Table 2.1: Standardized corpora that have been used for segmentation results reporting

Evaluation corpus	Description	Used in
Hub4-96 Evaluation [1]	2.5 hours, English broadcast news, Half consisting of anchored news broadcasts, other half from news magazines.	[Anguera, 2005], [Siegler *et al.*, 1997]
Hub4-97 Evaluation [2]	3 hours, English broadcast news Consists of anchored news shows, news magazines, hearings, news conferences and speeches.	[Chen & Gopalakrishnan, 1998b], [Tritschler & Gopinath, 1999], [Zhou & Hansen, 2000], [Vandecatseye & Martens, 2003], [Wu *et al.*, 2003], [Ajmera *et al.*, 2004], [Anguera, 2005]
RT-03s [3]	3 hours, English broadcast news Broken up into 6 x 30 minute shows.	[Meignier *et al.*, 2004], [Ajmera & Wooters, 2003], [Tranter & Reynolds, 2004], [Jin *et al.*, 2004]
COST278-BN [4]	30 hours, television broadcast news 9 European languages from 14 television stations.	[Zdansky *et al.*, 2004], [Zdansky & David, 2004], [Zibert *et al.*, 2005], [Teleki *et al.*, 2005]

[1] NIST 1996 Broadcast News Recognition Evaluation database http://www.nist.gov/speech/tests/bnr/1996

[2] NIST 1997 Broadcast News Recognition Evaluation database http://www.nist.gov/speech/tests/bnr/1997

[3] NIST Rich Transcription Spring 2003 Evaluation database http://www.nist.gov/speech/tests/rt/2003-spring

[4] COST278 Broadcast News Interest Group multi-lingual database https://speech.elis.ugent.be/s/index.php?Itemid=73

Chapter 3. This corpus has been frequently reported upon by various research teams for the broadcast news diarization task.

2.1.2.2 Scoring tolerance

As illustrated in Fig. 2.1, the transitions between speakers are often not precise. There may be some overlap between the speech of different speakers, or there may be an extended silence between them. This thus makes it impossible to put the turn point between two speakers down to an exact time-stamp. This problem is further exacerbated by the fact that the ground-truth references are often transcribed by a human. Human transcribers often have to exercise some judgment when determining exactly where a transition takes place. This judgment varies from transcriber to transcriber, and as such will introduce an element of variability into the references.

When scoring a set of segmentation results against a reference ground-truth, the most common way of coping with this issue is to use a scoring tolerance or "collar" around each reference point. As such, a hypothesized turn point does not have to coincide exactly with the ground-truth in order for it to be judged as correct. All it needs to do is to fall within the scoring tolerance of the reference ground-truth.

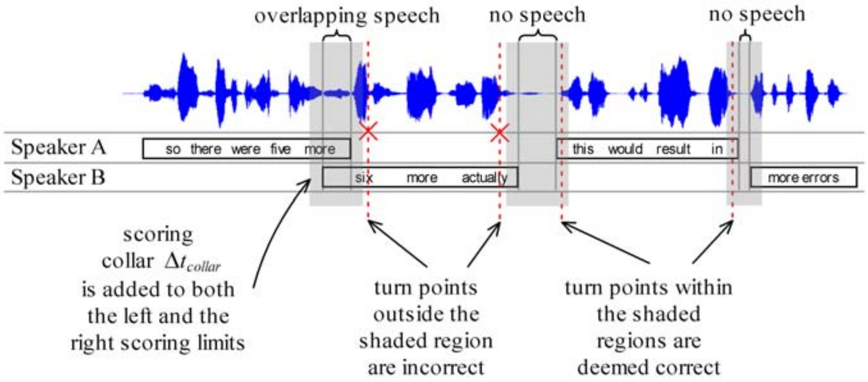

overlapping speech
no speech
no speech
Speaker A
so there were five more
this would result in
Speaker B
six more actually
more errors
scoring collar Δt_{collar} is added to both the left and the right scoring limits
turn points outside the shaded region are incorrect
turn points within the shaded regions are deemed correct

Figure 2.2: A scoring collar, Δt_{collar}, is used around each reference point. A hypothesized turn point will be deemed correct if it falls within the shaded regions.

The width of this scoring tolerance will thus have an impact on the evaluated segmentation performance. A larger tolerance will yield better performance measures than a smaller tolerance. Tolerance values of 2 seconds ($\Delta t_{collar} = 1$ second) have been used in papers such as [Chen & Gopalakrishnan, 1998b; Zhou & Hansen, 2000] while 3 seconds i.e. $\Delta t_{collar} = 1.5$ seconds was used by the paper [Kemp *et al.*, 2000]. A more stringent tolerance of 1 seconds ($\Delta t_{collar} = 0.5$ second) was adopted in [Ajmera *et al.*, 2004; Meinedo & Neto, 2003a]. The value of $\Delta t_{collar} = 0.5$ second has been adopted as the scoring standard for the COST278-BN Broadcast News evaluation exercises [Vandecatseye *et al.*, 2004].

2.2 Speaker Clustering

The speaker segmentation's objective was to break up a continuous recording into speaker homogeneous segments. Given this collection of segments, speaker clustering would usually then be applied in order to group the segments into clusters of common speakers. Application examples of the above include telephony mixed-channel speaker verification [Ore *et al.*, 2006; Deng *et al.*, 2006] and acoustic model adaptation for speech recognition [Pusateri & Hazen, 2002; Hain *et al.*, 2006; Janin *et al.*, 2006]. In the mixed-channel speaker verification task, speaker specific models after training are used to perform verification against a designated reference model. For the speech recognition task, the acoustic models used for recognition are adapted using the pooled speech. This would yield models specially tailored for the target speakers, thus translating into better overall recognition accuracy than when using generic models.

When performing speaker clustering, the decision on whether two segments should belong to the same cluster is often made using those algorithms also employed for speaker recognition or speaker segmentation. The vector quantization and model-based classification approach for clustering uses modeling techniques that have been employed for speaker recognition. The hierarchical clustering approach on the other hand uses divergence measures that are the same as those used in speaker segmentation.

2.2.1 Evaluation measures for clustering quality

When performing speaker clustering, there are two key concerns to the clustering quality. The following notations will be used when introducing the measures of clustering quality:

$N_{clusters}$: Number of clusters produced by the clustering algorithm.
N_{spk} : Actual number of speakers in the corpus according to the ground-truth scoring reference.
$n_{i,j}$: Number of audio frames in cluster i that is attributed to speaker j.
$n_{i,\bullet}$: Total number of audio frames in cluster i.
$n_{\bullet,j}$: Total number of audio frames attributed to speaker j across all clusters.
n_{all} : Total number of audio frames within the corpus,
i.e. $n_{all} = \sum_{i=1}^{N_{clusters}} n_{i,\bullet} = \sum_{j=1}^{N_{spk}} n_{\bullet,j}$

The first concern is the homogeneity of each resultant cluster. Every cluster should ideally consists of only a single speaker and be free from other contaminants. This homogeneity is usually measured using the cluster *Purity* [Gauvain *et al.*, 1998; Solomonoff *et al.*, 1998; Chen & Gopalakrishnan, 1998b; Tritschler & Gopinath, 1999; Cettolo, 2000].

The $Purity_i$ is defined as the proportion of audio frames in the i^{th} cluster originating from the most dominant speaker within that cluster. The dominant speaker is defined to be the speaker that has contributed the largest number of frames to the cluster. A higher *Purity* value thus represents greater homogeneity and the optimal will be $Purity = 1.0$ where every audio frame in the cluster is produced by a single speaker. For a given i^{th} cluster, the *Purity* is calculated by

$$Purity_i = \frac{1}{n_{i,\bullet}} \max_{j\in\{\text{all speakers}\}} \{n_{i,j}\} \tag{Eqn. 2.8}$$

The second concern regarding clustering quality is that the speech utterances originating from a single speaker be concentrated within as few clusters as possible. The extend of this speaker "concentration" is often measured using the *Coverage*. For a given j^{th} speaker, the *Coverage* is calculated across all clusters as

$$Coverage_j = \frac{1}{n_{\bullet,j}} \max_{i\in\{\text{all clusters}\}} \{n_{i,j}\} \tag{Eqn. 2.9}$$

When calculating the $Coverage_j$ for speaker j, the cluster containing the most number of frames for said speaker j is first identified. This is then expressed as a ratio of the total number of frames uttered by speaker j within the corpus. The optimal *Coverage* value thus is 1. A high *Coverage* value will mean a low dispersion level (or high concentration level) for the speaker.

2.2.2 Evaluation measures for the whole corpus

The *Purity* and *Coverage* measures that were described earlier measure the clustering performance for a single cluster or a single speaker. When comparing the performance between clustering systems, we are usually concerned about the performance across an entire corpus. The *Purity* and *Coverage* measures will thus serve as building blocks of the subsequent measures.

For a given corpus, given that there are $N_{clusters}$ number of clusters generated by a clustering system, the overall homogeneity of all clusters can then be represented using the Average Cluster Purity (ACP) [Gauvain *et al.*, 1998; Solomonoff *et al.*, 1998; Tritschler & Gopinath, 1999; Meinedo & Neto, 2003a; Vandecatseye & Martens, 2003; Tsai *et al.*, 2004; Barras *et al.*, 2006]. The ACP is the weighted mean of the *Purity* of every cluster. As such, just like *Purity*, a high ACP value close to 1 is desirable.

$$ACP = \frac{1}{n_{all}} \sum_{j=1}^{N_{clusters}} n_{\bullet,j} \cdot Purity_j \ \ (\%) \qquad \text{(Eqn. 2.10)}$$

In the same manner, by doing a weighted mean across the *Coverage* of all speakers, the Average Speaker Coverage (ASC) [Gauvain *et al.*, 1998; Barras *et al.*, 2006] can then be found. The ASC measures the dispersion of speech frames for all speakers. A high ASC value once again represents a low amount of dispersion and a better clustering algorithm.

$$ASC = \frac{1}{n_{all}} \sum_{i=1}^{N_{spk}} n_{i,\bullet} \cdot Coverage_i \ \ (\%) \qquad \text{(Eqn. 2.11)}$$

2.2.3 Estimating the number of speakers present

It is useful to bear in mind that in many clustering applications, the true number of speakers N_{spk} present in the corpus is unknown. N_{spk} will therefore have to be estimated and these clustering approaches often incorporate some means of determining N_{spk}. For hierarchical clustering approaches, this would usually take the form of having a stopping criterion, as is done in [Heck & Sankar, 1997; Betser *et al.*, 2004; Black & Schultz, 2006]. The purity-based halting of clustering can also be employed. This is done in works like [Siu *et al.*, 1992; Solomonoff *et al.*, 1998] where clusters that are deemed impure are split to form additional initial clustering states. In these approaches, the number of resultant clusters $N_{cluster}$ will thus serve as an estimate for N_{spk}.

2.3 Approaches to Speaker Segmentation

Numerous approaches to speaker segmentation have been introduced over the years. In this chapter and the next, these approaches have been roughly classified under 5 general categories.

i. Segmentation using silence

ii. Segmentation using divergence measures

iii. Segmentation by performing frame-level audio classification (See Section 2.5)

iv. Segmentation (and clustering) using a HMM decoder (See Section 2.5)

v. Segmentation (and clustering) using Direction of Arrival (See Section 2.5)

Approaches (i) & (ii) breaks up the continuous recording to a collection of unidentified segments. Speaker clustering will have to be used subsequently to identify and classify these segments into clusters. Approaches (iii)-(v) on the other hand performs speaker segmentation and clustering in a single integrated step.

While the primary purpose of speaker segmentation algorithms is to segment audio according to speaker identity changes, almost all of the segmentation methods that will be reviewed also respond to acoustic changes other than a change in the speaker's identity. What this means is that the algorithms do not strictly perform only speaker segmentation. Segmentation boundaries demarcating the introduction of music, or an acoustic transition to events such as a sneeze or clap can also be detected.

2.3.1 Segmentation using silence

The energy-based approach is perhaps the most elementary of all segmentation algorithms. It works by detecting parts of the continuous audio stream where the energy is lowest. These locations of low energy represent a pause in the continuous speech and thus can potentially indicate the transition between speakers. Segmentation is performed at these locations and further speaker clustering can then be done so as to determine the identity of the speakers present. This is the strategy that was used in papers such as [Siu *et al.*, 1992; Wegmann *et al.*, 1999b; Kemp *et al.*, 2000; Ore *et al.*, 2006]. In [Wegmann *et al.*, 1999b], an amplitude based silence detector is used as a first pass to break up continuous broadcast news recordings into segments.

The relative simplicity of the silence based segmentation approach also proves to be its weakness. As is mentioned in [Montaci & Caraty, 1998], many false turns will be generated. This can happen mid-sentence, when the speaker pauses in between words, or sometimes in the middle of words. Certain consonant such as the unvoiced frictatives are by nature usually of lower amplitude than others such as the vowels. An inopportunely placed frictative can thus result in an incorrect segmentation occuring midway of a word.

2.3.2 Segmentation using divergence measures

The basic idea underlying algorithms of this class is to run a sliding window across the entire continuous recording while computing the acoustic dissimilarity within the window. The sliding window will be divided into 2 parts - a left and a right sub-window. In between the sub-windows would be a hypothesized turn point. A divergence measure is computed between both sub-windows. This produces a metric that indicates the acoustic dissimilarity between both sub-windows. A segmentation of the audio can then be performed at locations where the divergence measure is at a maximum.

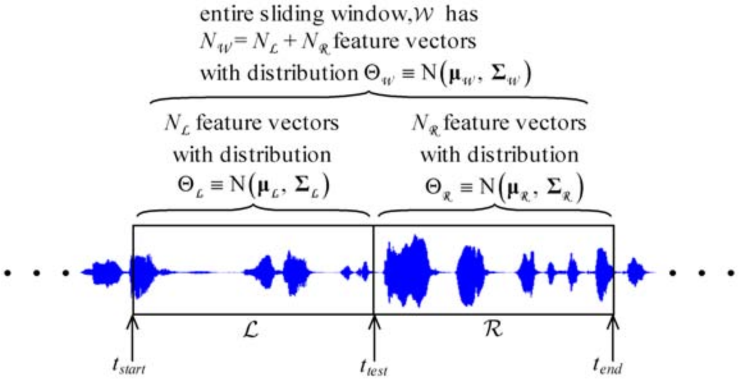

entire sliding window, $\mathcal{W}$ has
$N_{\mathcal{W}} = N_{\mathcal{L}} + N_{\mathcal{R}}$ feature vectors
with distribution $\Theta_{\mathcal{W}} \equiv N(\boldsymbol{\mu}_{\mathcal{W}}, \boldsymbol{\Sigma}_{\mathcal{W}})$
$N_{\mathcal{L}}$ feature vectors with distribution $\Theta_{\mathcal{L}} \equiv N(\boldsymbol{\mu}_{\mathcal{L}}, \boldsymbol{\Sigma}_{\mathcal{L}})$
$N_{\mathcal{R}}$ feature vectors with distribution $\Theta_{\mathcal{R}} \equiv N(\boldsymbol{\mu}_{\mathcal{R}}, \boldsymbol{\Sigma}_{\mathcal{R}})$
$\mathcal{L}$
$\mathcal{R}$
t_{start}
t_{test}
t_{end}

Figure 2.3: The sliding window used for divergence measure based algorithms.

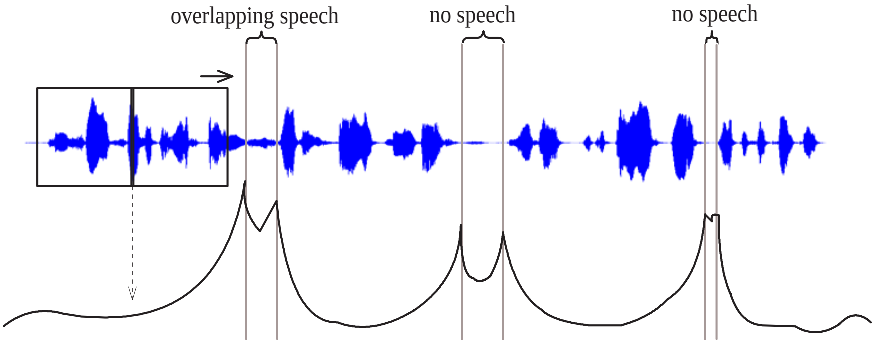


Figure 2.4: A time series plot of the divergence measure is obtained as the window is moved across the entire corpus. Maxima in the plot represent likely speaker turns.

The divergence between the windows is typically computed using audio feature vectors. Line Spectral Pairs (LSP) were the feature of choice in works such as [Lu & Zhang, 2002b]. The Mel-Frequency Cepstrum Coefficient (MFCC) [Davis & Mermelstein, 1980] however has been proven to be of the most popular feature for segmentation, as testified by the large number of works using it: [Tritschler & Gopinath, 1999; Kemp *et al.*, 2000; Delacourt & Wellekens, 2000; Wu *et al.*, 2003; Vandecatseye & Martens, 2003; Anguera, 2005; Doco-Fernndez & Garca-Mateo, 2005; Kim *et al.*, 2005] and many others. The Perceptual Linear Predictive Cepstral (PLPC) coefficients [Hermansky, 1990] is another popular feature that was used in works such as [Meinedo & Neto, 2003b; Wegmann *et al.*, 1999a].

An illustration of the typical sliding window regime is shown in Fig. 2.3. The left and right sub-windows of the sliding window will henceforth be denoted as $\mathcal{L}$ and $\mathcal{R}$. When this sliding window is moved across the entire corpus, a time-series plot of the divergence measure will be produced. Since a large divergence value typically represents a large acoustic difference between $\mathcal{L}$ and $\mathcal{R}$, the segmentation of the continuous audio can thus be done at the local maxima of the time-series. A threshold is usually also used as a criteria when detecting the maxima. Segmentation will take place only when the maxima exceeds the pre-determined threshold value.

In the computation of the divergence measures, it is usually assumed that the feature vectors from the windows $\mathcal{L}$ and $\mathcal{R}$ both have Gaussian distributions. Given an arbitrary segment $\mathbf{O}$ consisting of N audio feature vectors $\mathbf{o}_i$, $\{\mathbf{o}_i \in \mathbf{O} : i = 1 \dots N\}$, the likelihood probability that $\mathbf{O}$ belongs to either the $\mathcal{L}$ or $\mathcal{R}$ windows can thus respectively be defined:

$$\begin{aligned}
&\text{Probability that } \mathbf{O} \text{ belongs to } \mathcal{L},\ \ p_{\mathcal{L}}(\mathbf{O}) \\
&= p(\mathbf{O}|\Theta_{\mathcal{L}}) \\
&= \prod_{i}^{N} p(\mathbf{o}_i|\Theta_{\mathcal{L}}) \qquad \text{where}\, \Theta_{\mathcal{L}} \equiv \mathrm{N}(\boldsymbol{\mu}_{\mathcal{L}}, \boldsymbol{\Sigma}_{\mathcal{L}}) \qquad \text{(Eqn. 2.12)} \\
&\text{Probability that } \mathbf{O} \text{ belongs to } \mathcal{R},\ \ p_{\mathcal{R}}(\mathbf{O}) \\
&= p(\mathbf{O}|\Theta_{\mathcal{R}}) \\
&= \prod_{i}^{N} p(\mathbf{o}_i|\Theta_{\mathcal{R}}) \qquad \text{where}\, \Theta_{\mathcal{R}} \equiv \mathrm{N}(\boldsymbol{\mu}_{\mathcal{R}}, \boldsymbol{\Sigma}_{\mathcal{R}}) \qquad \text{(Eqn. 2.13)}
\end{aligned}$$

$\boldsymbol{\mu}_{\mathcal{L}}$ and $\boldsymbol{\mu}_{\mathcal{R}}$ respectively are the mean vectors of the $\mathcal{L}$ and $\mathcal{R}$ windows, while $\boldsymbol{\Sigma}_{\mathcal{L}}$ and $\boldsymbol{\Sigma}_{\mathcal{R}}$ are the covariance matrices. $\Theta_{\mathcal{L}}$ and $\Theta_{\mathcal{R}}$ respectively denote models of the $\mathcal{L}$ and $\mathcal{R}$ windows. The above notation will be used in the following Sections 2.3.2.1 & 2.3.2.2.

Table 2.2 lists some of the divergence measures that have been employed to perform speaker segmentation.

Table 2.2: A review of divergence measures used in speaker segmentation.

Divergence measure used System summary
Undirected Kullback-Leibler distance (KL2)
[Siegler *et al.*, 1997] Reported that the *KL2* could yield a *Recall* of 0.64, along with a *FAR* of 0.6. *KL2* was found to be less likely of making an erroneous decision about whether the windows $\mathcal{L}$ and $\mathcal{R}$ are from the same speaker. [Couvreur & Boite, 1999] Found that segmentation with the *KL2* could yield a lower *FAR* than that for Mahalanobis and Bhattacharyya distances, although the *Recall* for the latter two were better than that for *KL2*.
Divergence Shape Distance (DSD)
[Lu & Zhang, 2002a] Reported that the means of the distributions used in the *KL2* distance are easily biased by varying environmental conditions. As such, the *DSD* is used as an improvement over the *KL2* distance. The *DSD* measure was found to minimizes the effect of environmental or channel variations, and emphasizes the difference between speakers.
Mahalanobis distance (MAH) *Bhattacharyya distance (BHA)*
[Couvreur & Boite, 1999] Compared the *MAH* and *BHA* against the *KL2* divergence. It was found that the *Recall* was slightly better for *MAH* and *BHA* i.e. 0.935 & 0.966 respectively, versus 0.934 for *KL2* divergence

Continued on next page

Table 2.2 - continued from previous page

Divergence measure used	System summary
	[Hung *et al.*, 2000] Also reported that *Recall* was slightly better for *MAH* and *BHA* versus the *KL2* divergence.
Weighted Euclidean distance (WED)	[Kwon & Narayanan, 2002] Tested the *WED* against the *MAH* distance on a 1 hour recording of broadcast news. It was reported that the *WED* could deliver a 37.7% reduction in segmentation errors relative to the same for *MAH*.
Cross-Likelihood Ratio (CLR)	[Doco-Fernndez & Garca-Mateo, 2005] *CLR* was used after preliminary segmentation using the Bayesian Information Criterion (*BIC*). Reported that the *BIC* tends to be sensitive to situations where the acoustic environment changes but not the speaker present, resulting in a high number of false alarms. *CLR* was thus used as a confirmation step to ignore segmentations corresponding to acoustic changes, while retaining only those true speaker transitions.
Cross-BIC distance	[Anguera, 2005] Introduced as a simplification of the *CLR* distance as it was suggested that some terms in the *CLR* are redundant. Performance of *XBIC* was compared with that of *BIC* on the Hub4-97e corpus. Reported achieving an *FScore* of 0.637 versus 0.567 with *BIC*.
Generalized Likelihood Ratio (GLR)	[Bonastre *et al.*, 2000] Sliding windows are moved in fixed steps of 0.1 second, each half of the sliding window is 2 seconds. It was reported that over-segmentation tends to occur resulting is a high FAR. At a FAR of 0.50, a MDR of 0.15 was obtained.
	[Mori & Nakagawa, 2001] Reported that while the *GLR* expresses good speaker segregation capabilities, its overall performance was still slightly inferior to that of the *BIC*.
	[Remes *et al.*, 2007] Used the *GLR* to perform segmentation before performing clustering (or “speaker tracking”). A FAR of 0.51 and MDR of 0.16 was reported. The high FAR was not a problem because false speaker changes were resolved in the clustering phase.

The Bayesian Information Criterion (*BIC*) and Hotelling's T^2 statistic will be described to greater detail next as these two divergence measures will be used in Chapter 3 for speaker segmentation experiments.

2.3.2.1 Bayesian Information Criterion (BIC)

The divergence measure that is perhaps most often cited in speaker segmentation literature is the Bayesian Information Criterion (*BIC*) [Schwarz, 1978]. The *BIC* is a statistical criterion for model selection that has been used in papers such as [Chen & Gopalakrishnan, 1998b; Tritschler & Gopinath, 1999; Kemp *et al.*, 2000; Lopez & Ellis, 2000; Zhou & Hansen, 2000; Nishida & Kawahara, 2003; Vandecatseye & Martens, 2003; Ajmera *et al.*, 2004]. It essentially performs model selection by answering the following question about a window of speech surrounding a hypothesized turn point: “*Is this window best* *modeled using two distributions, or one single distribution ?"* The *BIC* equation penalizes a model's likelihood by the its complexity. All other things being equal, the *BIC* thus favours a model with lower complexity.

Fig. 2.3 illustrates the window notations used in the following derivation. Given that a model $\mathfrak{M}$ that is denoted by the statistical distribution Θ, the *BIC* for a window, $\mathcal{W}$, can be defined as

$$BIC(\mathfrak{M}) = \ln P(\mathbf{O}_{\mathcal{W}}|\Theta) - \frac{1}{2}\lambda D \ln N_{\mathcal{W}} \qquad \text{(Eqn. 2.14)}$$

$\mathbf{O}_{\mathcal{W}} = [\mathbf{O}_{\mathcal{L}}\ \mathbf{O}_{\mathcal{R}}] = [\mathbf{o}_{\mathcal{W},1} \ldots \mathbf{o}_{\mathcal{W},t_{test}} \ldots \mathbf{o}_{\mathcal{W},N_{\mathcal{W}}}]$ is the the series of $N_{\mathcal{W}}$ audio feature vectors captured within $\mathcal{W}$. D is the number of independent parameters present in Θ. The second term in Eqn. 2.14 is commonly referred to as the penalty term and is what penalizes the *BIC* score for its complexity. The model with the higher $BIC(\mathfrak{M})$ value thus is the model that should be chosen. For the purpose of segmentation, two models shall be defined:

- Model $\mathfrak{M}_0$ models the scenario where t_{test} is not a turn point. As such, the feature vectors for the left window ($\mathbf{O}_{\mathcal{L}}$) and right window ($\mathbf{O}_{\mathcal{R}}$) will belong a common distribution, $\Theta_{\mathcal{W}}$.

- Model $\mathfrak{M}_1$ on the other hand models the scenario where t_{test} is a turn point. As such, the feature vectors for the left window ($\mathbf{O}_{\mathcal{L}}$) and right window ($\mathbf{O}_{\mathcal{R}}$) will belong to two different distributions, $\Theta_{\mathcal{L}}$ and $\Theta_{\mathcal{R}}$.

It is assumed that the feature vectors $\mathbf{O}_{\mathcal{W}}$ follow a Gaussian distribution. As such, the likelihood of $\mathbf{O}_{\mathcal{W}}$, given $\Theta \equiv N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ would be

$$p(\mathbf{O}_{\mathcal{W}}|\Theta) = \prod_{i}^{N_{\mathcal{W}}} \left[\frac{1}{(2\pi)^{\frac{d}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{o}_{\mathcal{W},i}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{o}_{\mathcal{W},i}-\boldsymbol{\mu})} \right] \qquad \text{(Eqn. 2.15)}$$

$BIC(\mathfrak{M}_0)$ thus would be

$$\begin{aligned} BIC(\mathfrak{M}_0) &= \ln P(\mathbf{O}_{\mathcal{W}}|\Theta_{\mathcal{W}}) - \frac{1}{2}D_0 \ln N_{\mathcal{W}} && \text{(Eqn. 2.16)} \\ &= -\frac{d}{2}N_{\mathcal{W}} \ln 2\pi - \frac{N_{\mathcal{W}}}{2} \ln |\mathbf{\Sigma}_{\mathcal{W}}| - \frac{N_{\mathcal{W}}}{2} \\ &\quad -\frac{1}{2}\lambda \left(d + \frac{d(d+1)}{2}\right) \ln N_{\mathcal{W}} && \text{(Eqn. 2.17)} \end{aligned}$$

$D_0 = d + \frac{d(d+1)}{2}$ is the number of independent parameters present in $\Theta_{\mathcal{W}}$ and is an indication of the complexity for the model $\mathfrak{M}_0$. It consists of the d elements in the mean $\boldsymbol{\mu}_{\mathcal{W}}$ and the $\frac{d(d+1)}{2}$ unique elements in the covariance $\mathbf{\Sigma}_{\mathcal{W}}$. The covariance matrix is assumed to be diagonal in this case.

Similarly, the BIC value for $\mathfrak{M}_1$ is

$$\begin{aligned} BIC(\mathfrak{M}_1) &= \ln P(\mathbf{O}_{\mathcal{L}}, \mathbf{O}_{\mathcal{R}}|\Theta_{\mathcal{L}}, \Theta_{\mathcal{R}}) - \frac{1}{2}\lambda D_1 \ln (N_{\mathcal{L}} + N_{\mathcal{R}}) && \text{(Eqn. 2.18)} \\ &= \ln \left[P(\mathbf{O}_{\mathcal{L}}|\Theta_{\mathcal{L}})P(\mathbf{O}_{\mathcal{R}}|\Theta_{\mathcal{R}})\right] \\ &\quad -\frac{1}{2}\lambda \left(2d + d(d+1)\right) \ln (N_{\mathcal{L}} + N_{\mathcal{R}}) && \text{(Eqn. 2.19)} \\ &= -\frac{d}{2}(N_{\mathcal{L}} + N_{\mathcal{R}}) \ln 2\pi - \frac{N_{\mathcal{L}}}{2} \ln |\mathbf{\Sigma}_{\mathcal{L}}| - \frac{N_{\mathcal{R}}}{2} \ln |\mathbf{\Sigma}_{\mathcal{R}}| \\ &\quad -\frac{N_{\mathcal{L}} + N_{\mathcal{R}}}{2} - \frac{1}{2}\lambda \left(2d + d(d+1)\right) \ln (N_{\mathcal{L}} + N_{\mathcal{R}}) && \text{(Eqn. 2.20)} \end{aligned}$$

The D_1 independent parameters for this model consists of: d elements each in $\boldsymbol{\mu}_{\mathcal{L}}$ & $\boldsymbol{\mu}_{\mathcal{R}}$, and $\frac{d(d+1)}{2}$ elements in each covariance $\mathbf{\Sigma}_{\mathcal{L}}$ & $\mathbf{\Sigma}_{\mathcal{R}}$. Putting both $BIC(\mathfrak{M}_1)$ and $BIC(\mathfrak{M}_0)$ together, we can now formulate our hypothesis test using ΔBIC.

$$\begin{aligned} \Delta BIC &= BIC(\mathfrak{M}_1) - BIC(\mathfrak{M}_0) && \text{(Eqn. 2.21)} \\ &= -\frac{N_{\mathcal{L}}}{2} \ln |\mathbf{\Sigma}_{\mathcal{L}}| - \frac{N_{\mathcal{R}}}{2} \ln |\mathbf{\Sigma}_{\mathcal{R}}| + \frac{N_{\mathcal{W}}}{2} \ln |\mathbf{\Sigma}_{\mathcal{W}}| \\ &\quad -\frac{1}{2}\lambda \left(d + \frac{d(d+1)}{2}\right) \ln N_{\mathcal{W}} && \text{(Eqn. 2.22)} \end{aligned}$$

The null and alternate hypothesis for this model selection can be defined as

$$\begin{aligned} H_0 &: \Delta BIC \leq th_{peak} && \text{, there is not a turn point present at } t_{test} \\ H_1 &: \Delta BIC > th_{peak} && \text{, there is a turn point present at } t_{test} \end{aligned}$$

th_{peak} is the threshold value that determines whether the null hypothesis is to be rejected. A value of $th_{peak} = 0$ would be the fair threshold upon which to perform model selection. This value however can be adjusted to lend a bias to one of the models or hypothesis. The penalty factor λ is often tweaked in many papers so as to increase or decrease the complexity penalty. It has been reported in numerous papers [Tritschler & Gopinath, 1999; Lopez & Ellis, 2000; Vandecatseye & Martens, 2003; Ajmera *et al.*, 2004] that the ideal λ tends to be corpus dependent and λ thus is often selected by performing validation on a separate corpus.

One of the earliest papers to use the *BIC* for the task of speaker segmentation and clustering was [Chen & Gopalakrishnan, 1998b]. In that paper, the *BIC* was found to better predict a prospective turn point than the *KL2* and Gish distances. Analysis was performed on the divergence time-series produced by sliding a window across a 77 second speech recording containing a single speaker transition point. It was found that while the *KL2* and Gish distances showed local maxima peaks at the transition point, numerous spurious peaks were also observed that did not correspond to acoustic transitions. The *BIC* on the other hand produced only one single peak in the time-series plot and this peak corresponds perfectly to the transition location.

2.3.2.2 Hotelling's T^2 statistic

In the papers by Zhou & Hansen [Zhou & Hansen, 2000; Zhou & Hansen, 2005] and Huang & Hansen [Huang & Hansen, 2004; Huang & Hansen, 2006], a novel model selection approach using Hotelling's T^2 statistic was proposed. This method of using the T^2 statistic was meant to address one major shortcoming of the *BIC* algorithm, i.e. the *BIC* algorithm is computationally more complex and slow. The T^2 statistic on the other hand computes much quicker. This speed however comes at a price, the T^2 statistic method generally is less precise and thus needs to be coupled with the *BIC* algorithm in order to reduce its False Alarm Rate (FAR) and consequently improve its $Precision$.

The Hotelling's T^2 statistic is in effect a multi-variate generalization of the Student's t-statistic. It was introduced in 1931 by Harold Hotelling [Hotelling, 1931] and has found common usage in hypothesis testing. Given that a set of feature vector observations $\mathbf{O}$ is of Gaussian distribution,

$$\mathbf{O} \sim \mathrm{N}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \qquad \text{(Eqn. 2.23)}$$

The T^2 statistic to test if $\mathbf{O}$ indeed belongs to the distribution (i.e. whether $\hat{\boldsymbol{\mu}} = \boldsymbol{\mu}$) can be stated as

$$T^2(\mathbf{O}) = \mathrm{N}(\hat{\boldsymbol{\mu}} - \boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\hat{\boldsymbol{\mu}} - \boldsymbol{\mu}) \qquad \text{(Eqn. 2.24)}$$

$\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are the true population means and covariance respectively, while $\hat{\boldsymbol{\mu}}$ is the estimated sample mean. $T^2(\mathbf{O})$ measures the divergence of the sample observations from the population distribution. The smaller the value of $T^2(\mathbf{O})$, the closer the sample is to the population. Much like that for the *BIC*, the T^2 statistic can thus be used to formulate the segmentation problem in terms of a hypothesis test. For the purpose of the test, the feature vectors in the left and right windows are assumed to be normally distributed as $\mathbf{O}_{\mathcal{L}} \sim \mathrm{N}\left[\boldsymbol{\mu}_{\mathcal{L}}, \boldsymbol{\Sigma}_{\mathcal{L}}\right]$ and $\mathbf{O}_{\mathcal{R}} \sim \mathrm{N}\left[\boldsymbol{\mu}_{\mathcal{R}}, \boldsymbol{\Sigma}_{\mathcal{R}}\right]$.

The null and alternate hypothesis for the T^2 test can be stated as

$$\begin{aligned} H_0 \quad &: \hat{\mathbf{O}}_{\mathcal{L}} = \hat{\mathbf{O}}_{\mathcal{R}}, \quad \text{there is not a turn point present at } t_{test} \\ H_1 \quad &: \hat{\mathbf{O}}_{\mathcal{L}} \neq \hat{\mathbf{O}}_{\mathcal{R}}, \quad \text{there is a turn point present at } t_{test} \end{aligned}$$

$\hat{\mathbf{O}}_{\mathcal{L}}$ and $\hat{\mathbf{O}}_{\mathcal{R}}$ respectively are the sample means for the observations $\mathbf{O}_{\mathcal{L}}$ and $\mathbf{O}_{\mathcal{R}}$. The estimate of the sample means will thus have the distributions

$$\hat{\mathbf{O}}_{\mathcal{L}} \sim \mathrm{N}\left(\boldsymbol{\mu}_{\mathcal{L}}, \frac{\boldsymbol{\Sigma}_{\mathcal{L}}}{\mathrm{N}_{\mathcal{L}}}\right) \qquad \text{(Eqn. 2.25)}$$

$$\hat{\mathbf{O}}_{\mathcal{R}} \sim \mathrm{N}\left(\boldsymbol{\mu}_{\mathcal{R}}, \frac{\boldsymbol{\Sigma}_{\mathcal{R}}}{\mathrm{N}_{\mathcal{R}}}\right) \qquad \text{(Eqn. 2.26)}$$

The difference, $\mathbf{\Delta} = \hat{\mathbf{O}}_{\mathcal{L}} - \hat{\mathbf{O}}_{\mathcal{R}}$, will be distributed according to

$$\mathbf{\Delta} \sim \mathrm{N}\left(\boldsymbol{\mu}_{\mathcal{L}} - \boldsymbol{\mu}_{\mathcal{R}}, \frac{\mathbf{\Sigma}_{\mathcal{L}}}{\mathrm{N}_{\mathcal{L}}} + \frac{\mathbf{\Sigma}_{\mathcal{R}}}{\mathrm{N}_{\mathcal{R}}}\right) \qquad \text{(Eqn. 2.27)}$$

Assuming that $\mathbf{O}_{\mathcal{L}}$ and $\mathbf{O}_{\mathcal{R}}$ are from the same distribution,

$$\mathbf{\Sigma} = \mathbf{\Sigma}_{\mathcal{L}} = \mathbf{\Sigma}_{\mathcal{R}} \qquad \text{(Eqn. 2.28)}$$

$$\boldsymbol{\mu}_{\mathcal{L}} = \boldsymbol{\mu}_{\mathcal{L}} \qquad \text{(Eqn. 2.29)}$$

Eqn. 2.27 thus becomes

$$\mathbf{\Delta} \sim \mathrm{N}\left(0, \frac{\mathbf{\Sigma}}{\mathrm{N}_{\mathcal{L}}} + \frac{\mathbf{\Sigma}}{\mathrm{N}_{\mathcal{R}}}\right) \qquad \text{(Eqn. 2.30)}$$

The T^2 statistic used to test whether $\mathbf{O}_{\mathcal{L}}$ and $\mathbf{O}_{\mathcal{R}}$ are of the same distribution (i.e. whether $\boldsymbol{\mu}_{\mathcal{L}} - \boldsymbol{\mu}_{\mathcal{R}} = 0$) will be

$$\begin{aligned} T^2(\mathbf{\Delta}) &= [(\boldsymbol{\mu}_{\mathcal{L}} - \boldsymbol{\mu}_{\mathcal{R}}) - 0]^T \left(\frac{\mathbf{\Sigma}}{N_{\mathcal{L}}} + \frac{\mathbf{\Sigma}}{N_{\mathcal{R}}}\right)^{-1} [(\boldsymbol{\mu}_{\mathcal{L}} - \boldsymbol{\mu}_{\mathcal{R}}) - 0] \quad \text{(Eqn. 2.31)} \\ &= \frac{N_{\mathcal{L}} N_{\mathcal{R}}}{N_{\mathcal{L}} + N_{\mathcal{R}}} (\boldsymbol{\mu}_{\mathcal{L}} - \boldsymbol{\mu}_{\mathcal{R}})^T \mathbf{\Sigma}^{-1} (\boldsymbol{\mu}_{\mathcal{L}} - \boldsymbol{\mu}_{\mathcal{R}}) \quad \text{(Eqn. 2.32)} \end{aligned}$$

The critical region will be

$$T^2(\mathbf{\Delta}) > \frac{D_f - d + 1}{D_f \cdot d} F_{d, D_f - d + 1}(\alpha) \qquad \text{(Eqn. 2.33)}$$

where $D_f = N_{\mathcal{L}} + N_{\mathcal{R}} - 2$ is the number of degrees of freedom used in the test, d is the dimensionality of the observation vectors, and α is the significance level of the test. $F_{\bullet,\bullet}$ denotes a F-distribution [NIST, 2006a] with parameters as indicated in the sub-script. In other words, if Eqn. 2.33 is fulfilled, we shall reject H_0 and conclude that a turn point is present.

In the papers [Zhou & Hansen, 2000; Zhou & Hansen, 2005], the $T^2(\mathbf{\Delta})$ values is used as a first stage distance measure. The task of finding turn points will then become one of identifying peaks in the time-series curve of $T^2(\mathbf{\Delta})$. Hypothesized turn points from the first stage are then passed on to a second stage where the *BIC* measure is used to determine if the turn-points are false-positives. This T^2+*BIC* approach was reported to yield better FAR and MDR. A *BIC*-only system used for contrast produced $FAR = 10.8\%$ and $MDR = 29.3\%$. The two stage system on the other hand produced $FAR = 16.5\%$ and $MDR = 22.6\%$*. These results were reported upon the Hub4-97e corpus. Beyond the FAR and MDR results, the key performance difference between both system lies in the computational time required. The *BIC*-only system was timed to take 2160 minutes to complete segmentation while the T^2+*BIC* system took only 21 minutes. This works out to be a computational efficiency of about 100x faster for the two stage system.

2.4 Approaches to Speaker Clustering

For the task of clustering speech according to their speaker identities, the papers in the literature have mainly focused on the following 5 approaches:

i. Clustering using vector quantization

ii. Clustering using iterative model training and classification

iii. Clustering in a hierarchical manner using divergence measures

iv. Clustering (and segmentation) using a HMM decoder (See Section 2.5)

v. Clustering (and segmentation) using Direction of Arrival (See Section 2.5)

Approaches (i)-(iii) perform speaker clustering upon a set of unidentified speech segments. These approaches have their roots in speaker identification and indeed the problem underlying speaker clustering can be reduced to one of deciding whether two given speech utterances were made by a common speaker.

*The FAR and MDR results for the single stage *BIC*-only system, and the two stage T^2+*BIC* system have been converted to *Precision*, *Recall* and $FScore$ measures in Table 3.1. It can be seen that the two stage system is better with a $FScore$ of 0.803. The same for the single stage system is 0.789.

Approaches (iv) & (v) are combined segmentation and clustering approaches. Algorithms of these classes perform speaker segmentation and clustering together at the same time i.e. there are no distinct segmentation and clustering steps within the algorithms.

2.4.1 Clustering using Vector Quantization

Vector Quantization (VQ) has been explored for speaker clustering in papers such as [Cohen & Lapidus, 1995; Mori & Nakagawa, 2001; Akita & Kawahara, 2003; Rodrguez & Torres, 2004; Grebenskaya *et al.*, 2005; Haubold & Kender, 2006]. In the VQ based clustering approach, feature vectors from an unknown speaker are typically mapped to the known template vectors in a code-book. Each of these template vectors will represent a certain speaker identity. The mapping of feature vectors is performed using some distance measure and a decision about whether the speaker matches that in the template can be made by using the distance measure.

VQ clustering typically starts off with the creation of the code-book. This usually requires speech samples from every speaker and thus is not suited for applications requiring unsupervised clustering of the speakers. The most common way of initializing the code-book is probably using the Linde, Buzo & Gray (LBG) algorithm [Linde *et al.*, 1980]. This was done in works such as [Cohen & Lapidus, 1995; Akita & Kawahara, 2003; Rodrguez & Torres, 2004]. Self-Organizing Maps (SOM) [Kohonen, 1990] were also proposed in [Lapidot *et al.*, 2002; Lapidot, 2003] as a means of creating the code-book.

Each feature vector is then clustered to a speaker identity using some form of a distance measure. For this, the Vector Quantization distortion (VQD) measure is a commonly used measure of distance. The VQD was used in [Mori & Nakagawa, 2001; Kinnunen *et al.*, 2006; Haubold & Kender, 2006] and is defined using the Euclidean distance. A small VQD value thus represents greater similarity between the speech segment and a code-book template vector. It was observed in [Mori & Nakagawa, 2001] that the VQD exhibits robustness and reliability even for short speech utterances.

One key shortfall to the VQ-based clustering approach is the need for supervised training to create the code-book. This would mean that the method will not be able to performing clustering for speakers that were not enrolled in the training phase since these speakers will not have a corresponding template vector.

2.4.2 Clustering using iterative model training and classification

Model-based clustering approaches involving repeated training and classification were used with good results in papers such as [Deng *et al.*, 2006; Anguera *et al.*, 2006a]. The ability of the Gaussian Mixture Model (GMM) to model the identity of speakers is what underlies the algorithms of this class. The GMM has previously showed to be effective for speaker recognition in works such as [Reynolds & Rose, 1995; Scheffer & Bonastre, 2006]. By measuring the likelihood of speech segments against GMM speaker models, segments with high likelihoods can be assigned the identity of the GMM and those segments with similar identities can be clustered together. GMMs representing speakers are bootstrapped by iteratively re-training and re-classifying. Over many iterations, the segment assignments will stabilize to yield accurate representations of the different speaker clusters.

In [Anguera *et al.*, 2006a], an approach termed "Friends versus Enemies" was used for a meeting diarization system. A GMM $\Theta_{\mathcal{ALL}}$ is first trained using all the feature vectors in the entire meeting i.e. where $\mathcal{ALL} = \left\{\mathbf{O}_1 \ldots \mathbf{O}_{N_{segments}}\right\}$ represents the set of all segments in the meeting recording. The cross-likelihood for every segment is then obtained against $\Theta_{\mathcal{ALL}}$. A set of segments, $\mathcal{F}_0$, with the highest score are selected and termed as "friends" since they have a high level of similarity with $\Theta_{\mathcal{ALL}}$. A model $\Theta_{\mathcal{F}_0}$ is then trained using the segments from $\mathcal{F}_0$.

The remaining segments $\{\mathcal{ALL} - \mathcal{F}_0\}$ are then scored against $\Theta_{\mathcal{F}_0}$. The set of segments, $\mathcal{E}_0$, most dissimilar from $\Theta_{\mathcal{F}_0}$ will be termed as "enemies". A new model $\Theta_{\mathcal{E}_0}$

is then trained from $\mathcal{E}_0$ and whatever segments that remain are then scored against $\Theta_{\mathcal{E}_0}$. The algorithm reiterates and once again those segments with the highest similarity will be termed as "friends" and will belong to set $\mathcal{F}_1$. The same is repeated for set $\mathcal{E}_1$ to create yet another set of "enemies". This process of creating "friends" and "enemies" is repeated until all segments have been assigned to a set. The resultant GMMs $\Theta_{\mathcal{F}_0}, \Theta_{\mathcal{E}_0}, \Theta_{\mathcal{F}_1}, \Theta_{\mathcal{E}_1}, \ldots$ will then each represent a single speaker identity. This algorithm was reported to yield a good set of initial models for subsequent re-segmentation using HMM, especially when the number of speakers in the corpus is unknown.

2.4.3 Hierarchical clustering using divergence measures

As evidenced by the large amount of work that has been performed using it, hierarchical methods are a popular way of performing speaker and acoustic clustering. Hierarchical clustering in essence is a "Divide-and-conquer" algorithm [Webb, 2002]. It divides the task of clustering the speakers within a recording into a series of clustering sub-tasks. Each sub-task works upon a sub-set of the segments in the corpus.

Hierarchical methods can be divided into bottom-up methods or top-down methods. The difference between the two approaches are illustrated in Fig. 2.5. The methods essentially differ in whether they are agglomerative or partitional [Duda *et al.*, 2000]. Agglomerative algorithms start out with many segments. Segments are then successively clustered together, resulting in a few speaker clusters at the end point. Partitional clustering on the other hand start out with a single cluster containing every segment from the corpus. This big cluster is then successively broken up into smaller clusters. In both agglomerative or partitional clustering, the goal is the same i.e. to stop clustering where the number of clusters $N_{cluster}$ corresponds to the correct number of speakers N_{spk} present in the corpus.

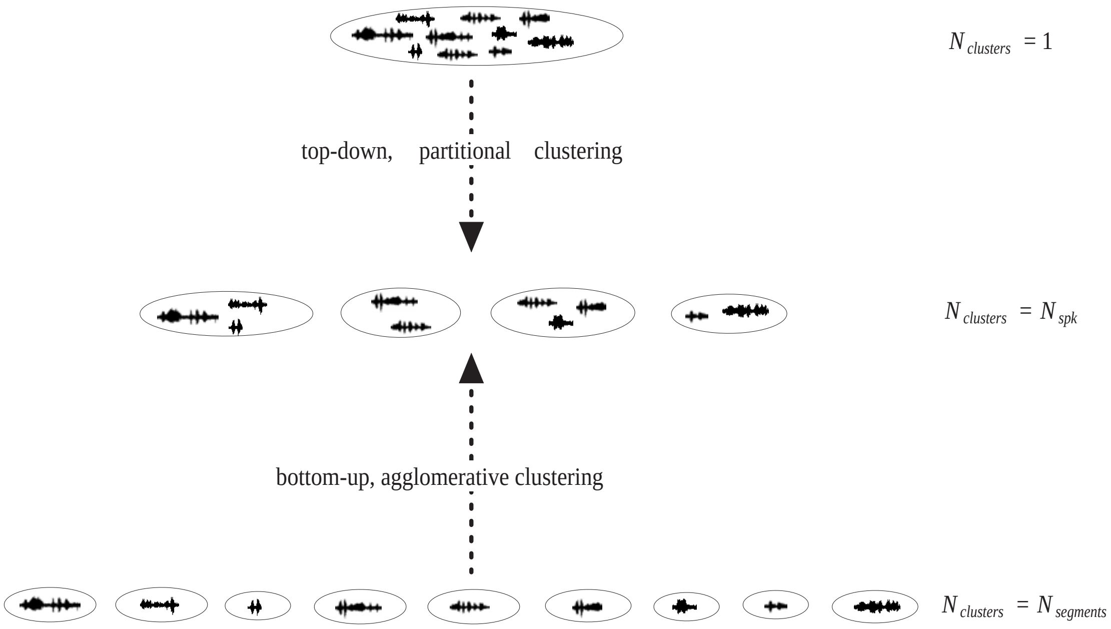

$N_{clusters} = 1$
top-down, partitional clustering
$N_{clusters} = N_{spk}$
bottom-up, agglomerative clustering
$N_{clusters} = N_{segments}$

Figure 2.5: Hierarchical clustering in a bottom-up or top-down fashion. The objective is to obtain a cluster number $N_{clusters}$ corresponding to the correct number of speakers N_{spk}.

2.4.3.1 Agglomerative bottom-up clustering

Agglomerative bottom-up clustering works by using a divergence measure to quantify the acoustic similarity between clusters of audio segments. A similarity matrix is typically constructed between all clusters under consideration for merging. The clusters that are most similar are then iteratively merged until when the stopping criterion is fulfilled. This criterion is often defined in the form of a threshold. In the event that the divergence measure exceeds or falls below the threshold value, clustering will halt and proceed to the next level in the hierarchy.

Table 2.3 lists some of the divergence measures that have been employed to perform bottom-up clustering. Included in the list are divergence measures such as the Kullback-Leibler (KL) distance, undirected Kullback-Leibler distance (i.e. $KL2$ distance), Cross Likelihood Ratio (CLR), Generalized Likelihood Ratio (GLR) and Bayesian Information Criterion (BIC) are distance measures which have been introduced previously for the purpose of segmentation in Section 2.3.2. Amongst these measures, the BIC is perhaps the most popular and commonly cited divergence measure. It has been used in papers such as [Tritschler & Gopinath, 1999; Zhou & Hansen, 2000; Vandecatseye & Martens, 2003] and has been reported to be capable of consistently yielding ACP measures above 95% on the Hub4-97e corpus.

Table 2.3: A review of agglomerative bottom-up clustering divergence measures and corresponding works.

Divergence measure used	
Systems description	Stopping criterion
Kullback-Leibler distance (KL)	
[Rougui *et al.*, 2006] Online indexing of speakers. Speakers are represented by GMMs, KL divergence is used to compute similarity between GMMs.	Not specified
[Harris *et al.*, 1999] Segments are weighted by proximity to each other. Segments near each other have priority.	Not specified
Undirected Kullback-Leibler distance (KL2)	
[Betser *et al.*, 2004] GMMs adapted for each segment from a background model, KL2 used to estimate distance between GMMs.	Squared Euclidean distance between representative feature vectors falls below th_{stop}.
Euclidean distance	
[Couvreur & Boite, 1999] Segments are mapped to a code-book using K-means, clustering is performed using code-book vectors.	Not specified
Cross-Likelihood Ratio (CLR)	
[Sankar *et al.*, 1995] Single, complete and average linkage results are examined.	Not specified
[Heck & Sankar, 1997] Average linkage is used where CLR score is an average of scores between all segments in both prospective merging candidates.	Stop when $CLR < th_{stop}$
[Barras *et al.*, 2004; Zhu *et al.*, 2006; Zhu *et al.*, 2005] CLR uses 3 GMMs - 2 to model each clustering candidate segment, 1 for a Universal Background Model (UBM). Score is normalized by size of candidate segments.	Stop when $CLR < th_{stop}$
[Nishida & Kawahara, 2005; Nishida & Kawahara, 2004] Clusters are modeled using GMM and VQ	Stop when $CLR < th_{stop}$
Generalized Likelihood Ratio (GLR)	
[Solomonoff *et al.*, 1998] Multi-level hierarchy was found to yield better clustering than a single-node hierarchy.	Threshold chosen where Average Cluster Purity (ACP) peaks.
[Lopez & Ellis, 2000] Distance measure used is a weighted combination of GLR and BIC.	Not specified
[Black & Schultz, 2006; Jin *et al.*, 2004; Jin & Schultz, 2004] GLR is used to compute distance, BIC is used to determine the stopping point.	Stop when $\Delta BIC < 0$
Bayesian Information Criterion (BIC)	
[Chen & Gopalakrishnan, 1998a] Clustering speakers for speech recognition model adaptation.	Stop when $\Delta BIC < 0$

Continued on next page

Table 2.3 - continued from previous page

Divergence measure used	
Systems description	Stopping criterion
[Tritschler & Gopinath, 1999] “Online” clustering where clustering hierarchy depends on segment order.	Stop when $\Delta BIC < 0$
[Zhou & Hansen, 2000] Gender classification is done first. Hierarchical clustering is then done within gender.	Stop when $\Delta BIC < 0$
[Meinedo & Neto, 2003a] BIC clustering results compared against the same when using KL2 were found to be slightly superior.	Stop when $\Delta BIC < 0$
[Moraru *et al.*, 2003] Segments are represented using GMMs MAP adapted from a background model, BIC compute distance between GMMs.	Stop at a pre-defined number of clusters
[Vandecatseye & Martens, 2003] ΔBIC score is normalized by the size of the candidate merging clusters.	Stop when normalized $\Delta BIC < 0$
[Kim *et al.*, 2005] HMM re-segmentation/clustering is done after BIC clustering	Stop when $\Delta BIC < th_{stop}$
Gish distance	
[Gish *et al.*, 1991] Clustering is performed on airport traffic control recordings.	Not specified
[Jin *et al.*, 1997] Heavier weighage were given to consecutive segments. Observations made that consecutive segments had higher likelihood of being from same speaker.	Clustering tree pruned to yield a pre-defined number of clusters.
Log-likelihood variant	
[Meignier *et al.*, 2002] Clustering results were compared against traditional CLR and found to do better.	Supervised clustering where dendrogram is pruned using a ACP based measure.
Speaker triangulation	
[Moh *et al.*, 2003] The similarity between each merger candidate speaker and all other speakers is found. This when summed together yields a measure that approximates to the GLR.	Not specified
Cosine distance	
[Tsai *et al.*, 2004] Feature vectors are projected onto a different dimensional space, cosine distance then used upon projected vectors.	Not specified
Earth Mover's distance (EMD)	
[Stadelmann & Freisleben, 2006] EMD is used in a MIXMAX framework, where a second GMM is used to model additive background noise.	Not specified

2.4.3.2 Partitional top-down clustering

Judging by the lesser amount of literature that is written about it, the partitional top-down hierarchical approach is much less popular than the agglomerative bottom-up speaker clustering method. The top-down hierarchical clustering technique starts by utilizing the speaker segments generated from an audio segmentation step. The segments are taken to form a single cluster at the start of the clustering hierarchy. This single cluster is then iteratively split into multiple clusters. Once clustering is complete, the end point will consist of clusters each corresponding to a speaker identity.

The Arithmetic Harmonic Sphericity (AHS) distance measure was proposed as a divergence measure for top-down clustering in [Bimbot & Mathan, 1993]. This distance measure was used in papers such as [Johnson & Woodland, 1998; Johnson, 1999]. In these systems, the division of segments within each clustering node occurs by iterating two steps. The first step is one where all segments are randomly assigned to one of 4 clusters. In the second step, the AHS distance of each individual segment with all 4 clusters is computed. The segment will then be assigned to the cluster in which it is deemed to be closest. The second step is iterated until such a point where the cluster assignments for all segments have stabilized. A minimum occupancy criterion is defined such that each of the 4 clusters must have a minimum number of segments assigned to it upon stabilization. Should this minimum occupancy be violated, the algorithm will re-start and repeat, this time working with only 3 clusters, as opposed to 4.

2.5 Joint Segmentation and Clustering Approaches

2.5.1 Segmentation by performing frame-level audio classification

Classifier-based approaches segment the continuous audio by performing some form of audio classification on the audio stream. The audio stream is classified in a frame-wise manner into various acoustic classes. Segmentation of the audio will then be done at the boundaries between audio of different classes. This ”segmentation-by-classification” approach is very similar to the decoder based approach that will be described in Section 2.5.2. It differs in that classifiers such as decision rules, Gaussian Mixture Models (GMM) or Support Vector Machines (SVM) [Burges, 1998] are used instead, as opposed to HMMs.

Table 2.4: A review of literature performing segmentation by frame-level audio classification.

Classifier used	System summary
Decision rules	[Zhang & Kuo, 1999] Classification is performed by applying decision rules on the characteristics exhibited by various acoustic features such as the signal zero crossing rate (ZCR), short-time energy, fundamental frequency and spectral peak track. The audio stream will be classified into one of eight classes: silence, harmonic environmental sound, non-harmonic environmental sound, pure speech, pure music, songs (i.e. speech sung along with music), speech against a music background, and environmental sound with music.
Nearest neighbour classifier	[Lu *et al.*, 2001b] Division of audio into speech and non-speech categories done using a K-nearest neighbour (KNN) classifier, then cascaded into a Linear Spectral Pair vector quantization (LSP-VQ) classifier for refinement. In the latter step, for each audio frame that is to be classified, the LSP distance of that frame is computed against template vectors in a codebook. The templates in this codebook represent speech vectors. Classification into speech and non-speech is then done depending on whether that LSP distance exceeds certain thresholds.
Gaussian Mixture Models (GMMs)	[Siu *et al.*, 1992] The subject of this paper was to label the speaker identities present in audio recordings between airport traffic controllers and pilots. The first step was to separate speech from noise. In this step, audio is first divided regularly into segments 200 milliseconds long. Each segment is then classified into speech or noise using a GMM classifier consisting of a speech and noise model. Noise boundaries are then used to segment the continuous audio and GMM models representing the speakers are then used to score each segment, providing speaker identities.
	[Tranter & Reynolds, 2004] Four GMM models were trained, one each for wide band speech, telephony speech, speech with music or noise, and pure music or noise. Speaker clustering was then performed using the models, scoring in a maximum likelihood manner. Comparisons were made between the GMM classifier based method and a traditional *BIC*-based segmentation and clustering. Holding all other things equal, the GMM based approach could yield better diarization error rates (*DER*). Tested upon the NIST Rich Transcription 2003 Spring (RT-03s) corpus, a *DER* of 32.2% was obtained with the GMM based approach while the *BIC* based approach yielded 33.91%.
Support Vector Machines (SVM)	[Lu *et al.*, 2001a] First pass segmentation was performed by detecting silence using the Short Time Energy (STE) feature. SVMs were then used to classify the non-silence portions into either music, speech or background noise. Experiments conducted upon a 2 hour corpus found that a classification accuracy of between 92% to 98% was obtained for the various audio classes. Also reported that the SVM based approach is computationally more efficient than the KNN method [Lu *et al.*, 2001b] previously described.

2.5.2 Segmentation and Clustering using a HMM decoder

Segmentation and clustering using a HMM decoder has proven to be a popular approach, especially when the final application is a speech recognition system. A HMM is typically used to decode the audio stream into acoustic classes. Recognition can then be carried out using models specific to that particular acoustic class. There however are variations in the decoder strategies used and Table 2.5 will give a summary of these strategies.

Table 2.5: A review of literature reporting segmentation and clustering using a HMM decoder.

System summary
[Hain *et al.*, 1998; Hain & Woodland, 1998] Classifies audio into the categories, with the added emphasis on determining if the speech is bandlimited (like telephony) or unlimited. Recognition is then carried out using acoustic models that matches the speech segment's acoustic quality, thus leading to better speech recognition results.
[Liu & Kubala, 1999] A HMM decoder is trained to detect gender changes and portions of non-speech in the continuous audio recording. This frame-by-frame information is then used to augment a Generalized Likelihood Rate (GLR) based segmenter where a dynamic threshold is used to decide if a segmentation should be done. The threshold is lowered at points where there the audio is deemed by the HMM decoder to be non-speech. Segmentations are also made where the decoder detected a change in the speaker gender. Results were obtained upon the Hub4-97e corpus [Graff *et al.*, 2002]. The ground-truth reference used had 483 turns, resulting in a FAR of 25% and a MDR of 30%. This would translate into $Recall$, $Precision$ and $FScore$ values of 0.7, 0.75 and 0.724 respectively.
[Meignier *et al.*, 2001] Because the number of speakers present in the corpus is unknown, a "top-down" approach was taken where the HMM starts off with 1 state (representing 1 speaker) and progresses towards having more states. The decoding process is repeated until where the number of speakers present is deemed to be optimal. During the decoding for each iteration, the HMM emission probability for each frame is stored. The frames that result in the highest occurrence likelihood within any single state are then taken to create a new HMM state. This process of adding a new state (each state representing a speaker) is repeated until when no more suitable frames are found.
[Ajmera & Wooters, 2003] The number of speakers present in the corpus is unknown and a "bottom-up" fashion is used with the HMM decoder. The number of speakers present in the speech is first hypothesized to be $\hat{N}_{spk}$. To initialize the decoder, the K-means algorithm is used to do a rough cluster of the audio frames into $\hat{N}_{spk}$ acoustic classes. A corresponding HMM state is trained for every acoustic class. Decoding is then performed and the two classes that are deemed to be most similar are merged. This is done using the GLR as the measure of similarity. The algorithm then reiterates. This is repeated until when the Viterbi decoding score is highest. This system tends to "under-cluster", i.e. the final number of speakers is greater than the actual number.

2.5.3 Segmentation and Clustering using Time Delay of Arrival

Given any time duration where an arbitrary speaker is speaking, since the multiple distant microphones (MDM) are placed in different locations within the room, the spatial distance between that speaker and each microphone will be different. The speed of sound is a constant and this would thus lead to differences in the times in which a sound arrives at each of the different microphones. This time differences would thus be the Time Delay of Arrival (TDOA) between the channels.

The general direction of arrival (DOA) of the speech signal can be interpreted using the TDOA, and consequently the location of speakers can be determined. The continuous audio recording can then be segmented when there is a change in the direction. Speaker clustering can also be performed by grouping together speakers originating from a common location. Table 2.6 offers a review of systems where the TDOA is used to perform segmentation and clustering of speech. It is notable that while all these systems performed TDOA estimation, there was no effort in any of them to accurately determine the direction of arrival in terms of the azimuth (i.e. the angle indicating the direction of arrival). To do so will require information about the microphone geometry and layout (information such as the exact microphone separation and orientation) and this is information that is not disclosed in the NIST Rich Transcription evaluation corpora.

Table 2.6: A review of literature performing segmentation and clustering using Direction of Arrival.

Method used for Time Delay of Arrival (TDOA) estimation
System summary
Generalized Cross-Correlation
[Ellis & Liu, 2004] 250 millisecond cross-correlation windows were used but larger windows were reported to yield more robust TDOA estimates. The TDOA estimates were then clustered to determine whether there was a change in speaker. An important finding in this paper was that some microphone pairs yield more reliable TDOA results and a process of microphone pair selection will be useful. The issue of "good peak" selection was also raised and the authors purposed applying spectral clustering on the audio signal to help remove spurious peaks. Diarization results were reported upon the NIST RT-04 Spring corpus - a DER of 62.3% was obtained.
[Wooters *et al.*, 2004] The TDOA estimates were used only to enhance audio recordings from multiple distant microphones (MDM). For every discrete utterance, the cross-correlation is used to estimate the delays between channels. Delay-and-summing was then performed to yield a single enhanced channel which was used for speech recognition. A 6.6% speech recognition word error rate improvement was observed between the enhanced channel and the best MDM recording.
Generalized Cross-Correlation using Phase Transform (GCC-PHAT)
[Cheng *et al.*, 2005] The cross-correlation and GCC-PHAT methods of obtaining TDOA were compared in a speaker segmentation task. It was reported that cross-correlation worked better than the GCC-PHAT. The authors acknowledged that this finding goes against the findings of numerous other works [Anguera *et al.*, 2005b; Chen *et al.*, 2006] suggesting that the GCC-PHAT works better than the cross-correlation method in reverberant environments, and offered that this contradiction occurred because the GCC-PHAT algorithm requires a larger window size than the 32 millisecond that was used.
[Anguera *et al.*, 2005b] TDOA estimates were used as a feature alongside acoustic features for used in a traditional segmentation and clustering algorithm using the BIC divergence measure. The TDOA was first estimated using the GCC-PHAT algorithm. The TDOA features were then weighted and combined with acoustic features, forming common feature vectors. Segmentation was then performed on these feature vectors using the BIC measure, and the resultant segments likewise were clustered using BIC. It was found that the appropriate weighting of the TDOA features were essential because when correctly weighted using an automatic process, a DER improvement of 18.2% could be obtained against that using manual weights.
[Pardo *et al.*, 2006] TDOA estimates are used in the paper to perform speaker segmentation and clustering, in lieu of regular acoustic

Continued on next page

Table 2.6 - continued from previous page

Method used for Time Delay of Arrival (TDOA) estimation
System summary
features. A speech versus non-speech detector was first used to remove portions of the audio recordings that are non-speech. TDOA estimation was then done using GCC-PHAT on the portions labeled speech. The sliding windows used were 500 millisecond long. The TDOA estimates from the many MDM pairs were then fed into a HMM decoder. The HMM decoder was initialized by equally dividing the continuous audio into K segments, each segment was used to train a HMM state. K is selected to be larger than the actual number of speakers present. An iterative process of decoding and merging then ensues. Diarization results were reported upon the NIST RT-04 Spring corpus - a DER of 33.67% was obtained.

The Normalized Least Means Squared (NLMS) filter is another method that can be used to estimate the TDOA across channels. This method was first proposed in [Reed *et al.*, 1981] and has been used in various forms in papers such as [Youn *et al.*, 1982; Chen *et al.*, 2006]. To the best of our knowledge however, this method of estimating TDOA has never been used in the context of performing speaker diarization, segmentation or clustering. It was reported in [Chen *et al.*, 2006] that a key advantage of this method would be its low computational complexity. The NLMS filter method however tends to perform poorly in the presence of reverberations, or when the signal is especially noisy. This method will nevertheless be employed in the system detailed in Section 4.3 because of its ease of implementation.

2.6 Summary of this chapter

This chapter reviewed published techniques for segmentation and clustering in the speaker diarization task. The Bayesian Information Criterion (BIC) was explored to a greater detail because it is perhaps the divergence measure that is most often used for the purpose of speaker segmentation and hierarchical clustering. The BIC is still used in current literature as a reference algorithm and experiments will be conducted in the next Chapter repeating the results reported in [Ajmera *et al.*, 2004] on the Hub4-97 Evaluation Broadcast News corpus.

The Hotelling's T^2 divergence measures was also explored to a greater detail as it appears to be a promising complement for the BIC. It was reported in works primarily by Zhou & Hansen [Zhou & Hansen, 2000; Zhou & Hansen, 2005] and Huang &

Hansen [Huang & Hansen, 2004; Huang & Hansen, 2006] to be good for performing a first pass shortlisting before applying BIC because it is computationally less complex. An attempt shall be made in the next chapter to replicate the experiments carried out in [Zhou & Hansen, 2005].

A review of the GMM-based iterative model training and clustering technique was conducted with a particular interest in [Anguera *et al.*, 2006a]. The iterative GMM-based clustering approach would be implemented later in Chapter 4 for the purpose of "cluster purification".

This Chapter finally ends with a review of the literature describing where Direction of Arrival has been used specifically for the purpose of performing speaker segmentation and clustering. A Normalized Least Means Squared (NLMS) filter based Time Delay of Arrival (TDOA) estimation module will be developed as the front-end for the diarization system in Chapter 4

Chapter 3

Experiments on the Hub4-97 Corpus

In this chapter, we describe experimental results conducted using the Hub4-97 broadcast news corpus to evaluate both segmentation and clustering performance. We first describe the database, then report on results presented by other researchers for this database, and finally presented our own experimental work repeating published works on segmentation.

3.1 The Hub4-97 Evaluation Broadcast News Corpus

The 1997 Hub4 English Evaluation corpus [Graff *et al.*, 2002] (Hub4-97e) was originally released by NIST in 1997 for the evaluation of speech recognition technology on English broadcast news. It consists of almost 3 hours of English broadcast news, has 555 segments of which 91 of these segments contained non-speech audio such as advertisement consisting of sound effects and music, or background noise. The average segment length is 19.1 seconds and 93 of these segments are of length 2 seconds or less. There are 117 unique speakers in the corpus, of which 31 are female and 86 are male.

3.1.1 Segmentation Results Reported on the Hub4-97e Corpus

One of the earliest papers to report speaker segmentation on the Hub4-97e was [Chen & Gopalakrishnan, 1998b]. That paper introduced the idea of performing segmentation and clustering using the *BIC* model-selection approach. The authors reported a false alarm rate (FAR) of 4.1% and a Missed Detection Rate (MDR) of 33.4%. In 1999, [Tritschler & Gopinath, 1999] introduced a variable window scheme for the *BIC* segmentation approach. Their experiments were also conducted upon the Hub4-97e and they reported a FAR of 9.2% with a MDR of 24.7%. This paper also performed clustering using *BIC* as the distance measure. Their system was capable of producing 149 clusters of 98.58% purity.

As the *BIC*'s computational complexity is high, [Zhou & Hansen, 2000] introduced a two-pass hybrid approach which first analyzed the data using the computationally simpler Hotelling's T^2 statistic followed by *BIC*. Their hybrid approach ran 100 times faster than using just plain *BIC* and they reported a FAR of 16.5% with a MDR of 22.6%. Clustering was also done and the reported cluster $Purity$ was 99.3%.

A more recent work citing the Hub4-97e corpus is [Ajmera *et al.*, 2004] in 2004. In that paper, log-likelihood ratios (LLR) between GMMs models were used to perform segmentation. Performance comparisons between this LLR approach and the *BIC* approach were made and it was shown that while the *BIC* could potentially outperform the LLR, this required the tuning of the *BIC* penalty factor λ. The LLR approach however has the advantage that it did not require any explicit thresholds and as such no tuning was required. The *Recall* and $Precision$ results reported were 0.65 and 0.68 respectively were reported for the LLR approach, and the best *BIC* performance had a *Recall* of 0.71, with a $Precision$ of 0.66.

Table 3.1 summarizes those segmentation scores that have been reported on the Hub4-97e corpus. Note that column 2 of the table indicated the number of turn points specified by each author. In summary, we note that different authors have chosen different scoring tolerances and different number of speaker turn points. As such, it is difficult to draw concrete conclusions from the reported experimental results. We will hence conduct our own implementation of reported algorithms in order to measure their performance in the following section.

Table 3.1: Segmentation results reported on the Hub4-97e corpus

System	Turn points considered	Scoring tolerance Δt_{collar} (seconds)	*FAR* (%)	*MDR* (%)	*Precision*	*Recall*	*FScore*
[Chen & Gopalakrishnan, 1998b]							
BIC	620	1	4.1	33.4	(0.959)	(0.666)	(0.786)
[Tritschler & Gopinath, 1999]							
BIC	not stated	not stated	9.2	24.7	(0.908)	(0.753)	(0.823)
[Liu & Kubala, 1999]							
BBN [1]	620	not stated	56.3	49.2	(0.437)	(0.508)	(0.470)
CMU [1]	620	not stated	64.1	42.8	(0.359)	(0.572)	(0.441)
normalized *GLR*	620	not stated	25.0	30.0	(0.750)	(0.700)	(0.724)
ditto, speech turns only	482	not stated	20.0	29.5	(0.800)	(0.705)	(0.750)
[Zhou & Hansen, 2000]							
BIC only	548	1	10.8	29.3	(0.892)	(0.707)	(0.789)
T^2+*BIC*	548	1	16.5	22.6	(0.835)	(0.774)	(0.803)
[Vandecatseye & Martens, 2003], normalized *BIC*							
speaker turns only	515	not stated	(25.8)	(20.9)	0.742	0.791	0.766
all acoustic changes	624	not stated	(34.6)	(17.6)	0.654	0.824	0.729
[Wu *et al.*, 2003]							
DSD [2]	not stated	not stated	33.8	10.8	(0.662)	(0.892)	(0.760)
[Ajmera *et al.*, 2004]							
BIC	515	0.5	(29.0)	(34.0)	0.710	0.660	0.684
GMM-LLR	515	0.5	(35.0)	(32.0)	0.650	0.680	0.665
[Anguera, 2005]							
BIC	512	not stated	39.0	47.0	(0.610)	(0.530)	(0.567)
XBIC	512	not stated	35.0	37.0	(0.650)	(0.630)	(0.637)

(·) Figures enclosed in brackets were not provided by the respective authors and were thus calculated.

[1] These are not results directly reported by BBN & CMU per se. Rather, they were reported in [Liu & Kubala, 1999] but attributed to BBN & CMU.

[2] DSD: Divergence Shape Distance

3.2 Speaker segmentation experiments on the Hub4-97e corpus

This section describes the speaker segmentation experiments that were carried out to replicate results reported in the literature. The *BIC* and T^2 model-selection based distance measures were selected for testing. The *BIC* model-selection approach was selected because of the large number of publications reported using it, as well as the good performance results reported in those publications. It has been reported to be superior to the *GLR* in [Mori & Nakagawa, 2001], and was observed in [Chen & Gopalakrishnan, 1998b] to have better turn point discriminating characteristics than the *KL2* and Gish distances. We will also examine the *BIC* segmentation system and the effect of the penalty factor λ.

The T^2 was also selected for experimentation on the basis of results reported in [Zhou & Hansen, 2005]. In those papers, the T^2 with its high *Recall* (low MDR) and speed was reported to be a good complement for the *BIC*. The low *Precision* (high FAR) can then be improved by using *BIC* in a second pass to reduce the number of false turns. The following experiments will thus be conducted upon the T^2 segmentation system:

(i) Finding the optimal th_{peak} for T^2.

(ii) Examining the results from the fusion of T^2 with *BIC*.

3.2.1 Experimental setup

The system used for the experiments was developed in Matlab. This system can be divided into three modules - the Feature Extraction module, the Segmentation module and then the Scoring module.

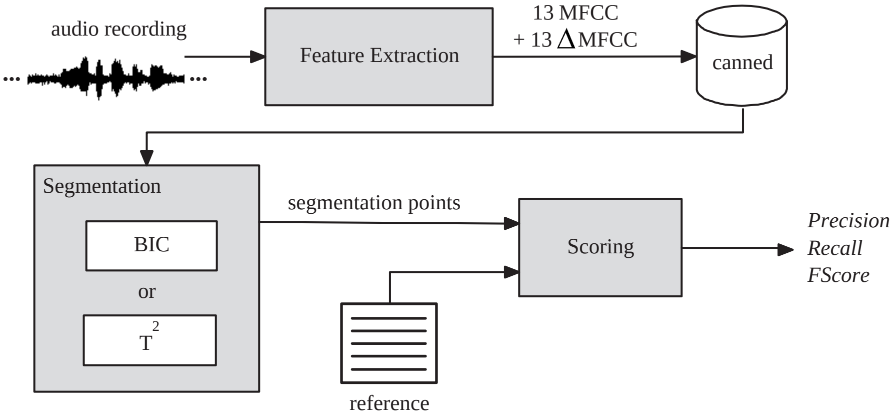


Figure 3.1: Block diagram of the speaker segmentation system used in the experiments.

In the Feature Extraction module, the system reads in the input Hub4-97e audio in the form of Microsoft PCM .wav files. The input audio was sampled at a 16kHz rate, with a sample resolution of 16-bits. 13 Mel-Frequency Cepstrum Coefficient (MFCC) features [Davis & Mermelstein, 1980] and their deltas were extracted from each audio frame. Each resultant feature vector thus is 26 dimensional. A frame size of 30 ms was used and the frame hop is 30 ms. There is thus no overlapping between consecutive frames. In order to reduce signal discontinuities at the boundary between frames, the samples within each frame are weighted using a Hamming window [Hamming, 1989]. The generated features are then stored and passed on to the Segmentation module. Both the BIC and T^2 segmentation algorithms thus share a common feature input in the form of the stored data.

The Segmentation module then performs turn point detection using a two-part sliding window algorithm as described in Section 3.2.2. The parameters specific to the BIC or T^2 algorithms were varied and the effect of their variation on the resultant segmentation was studied. With each variation in the parameters, a time series plot of the ΔBIC or $\mathrm{T}^2(\boldsymbol{\Delta})$ score for the whole corpus will be obtained. The determination of turn points is done by detecting local maxima in the plot.

The turn point detection results from the Segmentation module are then passed onto the Scoring module for evaluation. In the Scoring module, the detection results are compared against a set of ground-truth turn points for scoring. A scoring tolerance of 2 seconds (Δt_{collar} = 1.0 seconds) is used to determine if a turn point is correct. Each hypothesized turn point is then deemed to be either correctly identified, or an insertion error (false alarm). This depends on whether it falls within the scoring region corresponding to a ground-truth turn point. In the event that two or more hypothesized turn points fall into the region corresponding to a ground-truth, only the most accurate hypothesized points will be taken to be correct. The others will be deemed as insertion errors. After all the hypothesized turn points have been scored, the ground-truth reference is then evaluated for deletion errors. Ground-truth turn points that do not have a hypothesized point coinciding with them will be regarded as deletion errors.

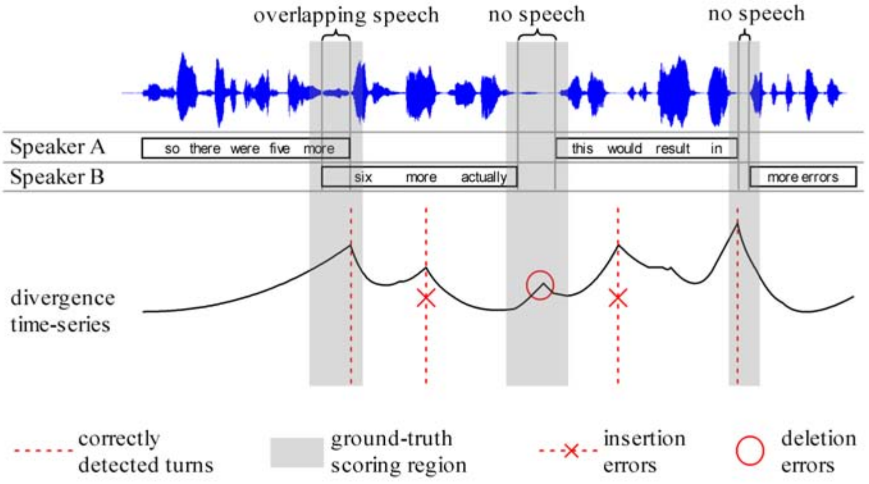


Figure 3.2*: The scoring criteria for correctly detected turns, insertion errors and deletion errors.*

With the number of correct turns, insertion and deletion errors determined, the performance of the system will be presented using the $Recall$, $Precision$ and $FScore$ measures.

3.2.2 The windowing algorithm used

The windowing algorithm used is adapted from [Ajmera *et al.*, 2004]. The key idea behind the algorithm is to keep growing the window until a prospective turn point is found (Fig. 3.3). When a prospective turn point is found, the window resets by sliding itself to start at said turn point.

In the following algorithm description, $divergence(t_{start}, t_{test}, t_{end})$ represents the computation of the divergence between the $\mathcal{L}$ and $\mathcal{R}$ sub-windows of an instance of the sliding window at time t_{test}. This divergence can be computed using any algorithm, be it the *BIC* or the T^2. $v[t]$ will be the variable used to store the divergence measure computed and thus contains the time-series plot of the divergence.

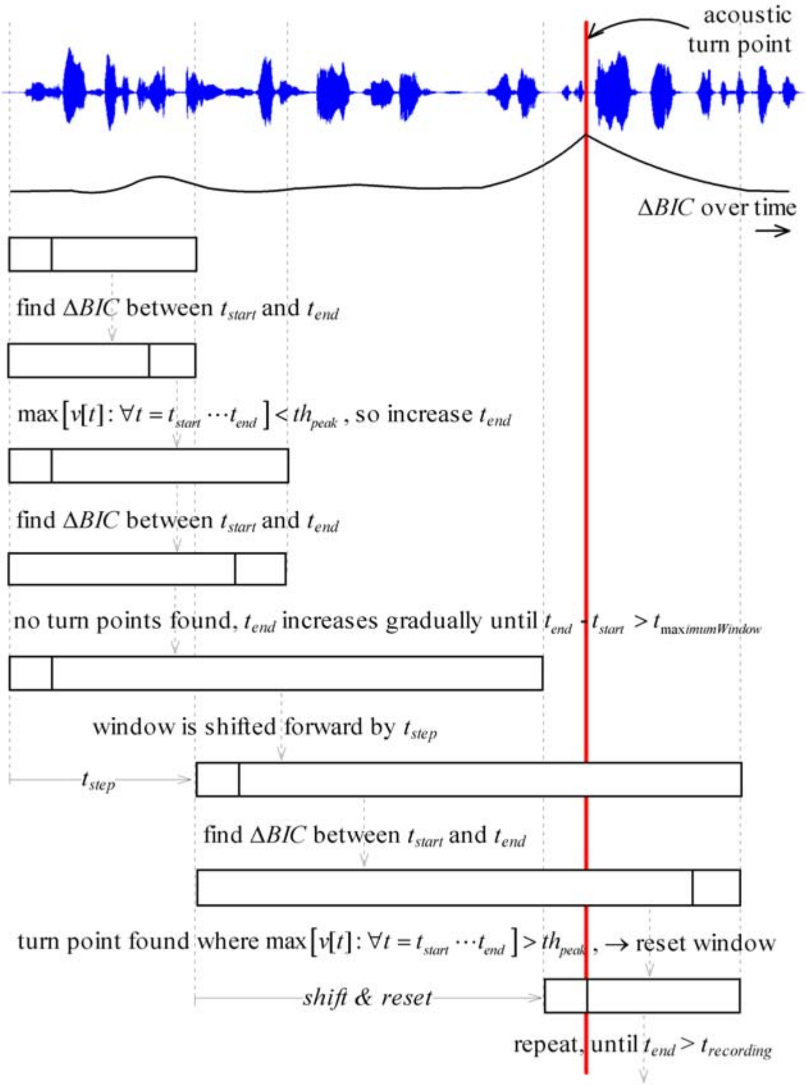

acoustic
turn point
ΔBIC over time
find ΔBIC between t_{start} and t_{end}
$\max\left[v[t] : \forall t = t_{start} \cdots t_{end}\right] < th_{peak}$, so increase t_{end}
find ΔBIC between t_{start} and t_{end}
no turn points found, t_{end} increases gradually until $t_{end} - t_{start} > t_{\max imumWindow}$
window is shifted forward by t_{step}
t_{step}
find ΔBIC between t_{start} and t_{end}
turn point found where $\max\left[v[t] : \forall t = t_{start} \cdots t_{end}\right] > th_{peak}$, → reset window
shift & reset
repeat, until $t_{end} > t_{recording}$

Figure 3.3: The windowing algorithm used for divergence computation. Although the BIC is illustrated here, the same algorithm is also used for the T^2.

```
Windowing algorithm for both the $BIC$ and $\mathrm{T}^2$ systems:

  (i) Initialize $v[t] = 0$, $\forall t = 0 \cdots t_{recording}$ seconds

 (ii) Let
        $t_{start} = 0$ seconds
        $t_{end} = t_{start} + t_{minimumWindow}$ seconds

(iii) For $t_{test} = t_{start} + t_{minimumLeft}$ to $t_{end}$ - $t_{minimumRight}$
        $v[t_{test}] = \max\left[v[t_{test}], divergence(t_{start}, t_{test}, t_{end})\right]$
      end

 (iv) If $\max\left[v[t] : \forall t = t_{start} \cdots t_{end}\right] > th_{peak}$ then
        $t_{start} = \arg\max\left[v(t) : \forall t = t_{start} \cdots t_{end}\right] - t_{minimumLeft}$ seconds
        $t_{end} = t_{start} + t_{minimumWindow}$ seconds
      else
        $t_{end} = t_{end} + t_{step}$ seconds
        if $t_{end}$ - $t_{start} > t_{maximumWindow}$ then
          $t_{start} = t_{start} + t_{step}$ seconds
        end
      end

  (v) Repeat from (iii) until $t_{end} > t_{recording}$
```

3.2.3 Detecting local maximums in the divergence time-series

After passing the sliding window across the entire recording, the divergence measured at every point t can be plotted using the time-series $v[t]$ (Fig. 3.4). The detection of the local maximums (and correspondingly acoustic turn points) is done by using a second sliding window scheme. This sliding window is moved across $v[t]$ and each time the contents of the window fulfills the following criteria, a segmentation is made. The span of this second sliding window is t_{span}.

A segmentation occurs if $v[t]$, the divergence result at time t fulfills

(i) $v[t] > th_{peak}$

(ii) $v\left[t-1\right] < v\left[t\right]$

(iii) $v\left[t\right] > v\left[t+1\right]$

(iv) $t = \underset{t-\frac{t_{span}}{2} \leq i \leq t+\frac{t_{span}}{2}}{\arg\max} \{v[i]\}$

The purpose of criterion (ii) & (iii) is to ensure that $v[t]$ is a peak in its immediate vicinity i.e. that the values immediately left and right of $v[t]$ are both smaller than $v[t]$. Criterion (iv) ensures that there can only be one peak found within the locality defined by t_{span}. A value of $t_{span} = 1$ second is used in the subsequent experiments.

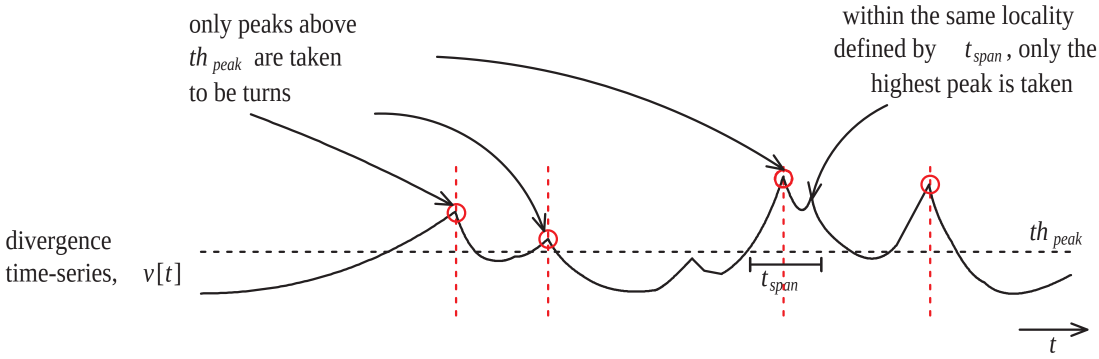


Figure 3.4: The criteria used for finding turn points in divergence time series.

3.2.4 Segmentation using the Bayesian Information Criterion

The ΔBIC equation used is stated again as follows. This is similar to what was derived earlier in Eqn. 2.22.

$$\begin{aligned} \Delta BIC &= -\frac{N_{\mathcal{L}}}{2} \ln |\mathbf{\Sigma}_{\mathcal{L}}| - \frac{N_{\mathcal{R}}}{2} \ln |\mathbf{\Sigma}_{\mathcal{R}}| + \frac{N_{\mathcal{W}}}{2} \ln |\mathbf{\Sigma}_{\mathcal{W}}| \\ &\quad -\frac{1}{2}\lambda \left(d + \frac{d(d+1)}{2} \right) \ln N_{\mathcal{W}} \end{aligned} \qquad \text{(Eqn. 3.1)}$$

Since 13 MFCC features and their corresponding ΔMFCC are used, each feature vector would be d =26 long. In the subsequent experiments, the value of the turn detection threshold th_{peak} will be held constant at $th_{peak} = 0$. The parameter λ will then be used as a performance tuning parameter.

3.2.4.1 The effect of the penalty factor λ

The fact that the optimum penalty factor λ for ΔBIC varies from corpus to corpus has been mentioned in numerous papers such as [Tritschler & Gopinath, 1999; Lopez & Ellis, 2000; Vandecatseye & Martens, 2003; Ajmera *et al.*, 2004]. A value of λ that yields a good score on one corpus thus may not have the same positive effect on a second corpus. Fig. 3.5 shows the ΔBIC curves for values of $\lambda = 0.5$ and $\lambda = 1.0$. These two curves will

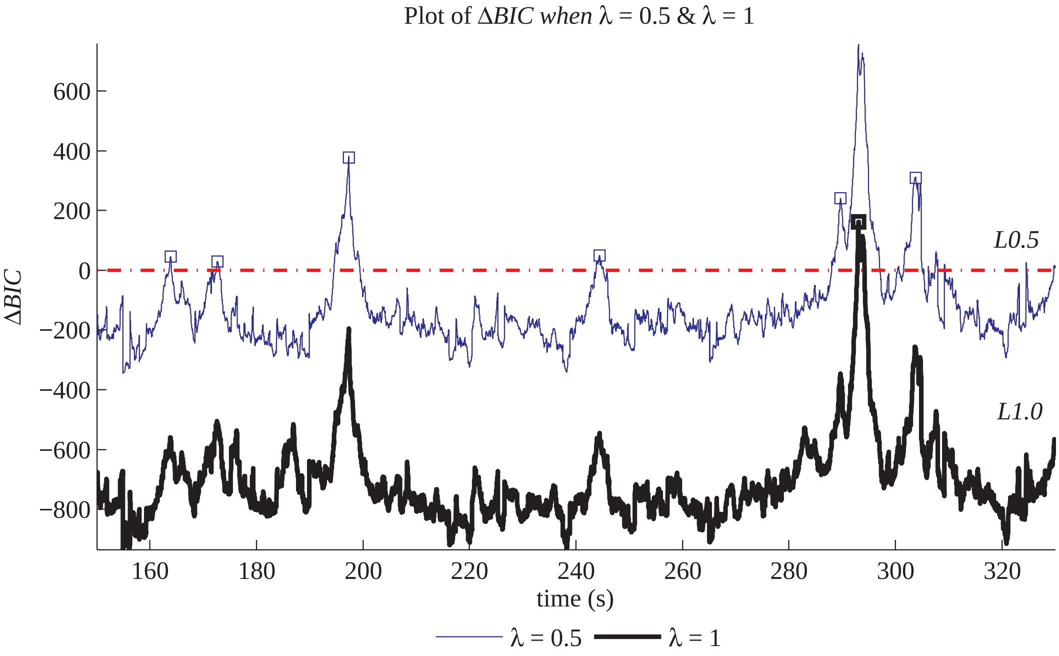


Figure 3.5: For a segment of audio from Hub4-97e, time series plot of ΔBIC computed using $\lambda = 0.5$ versus $\lambda = 1.0$. Square markers (□) indicate speaker turns detected. Notice the plots are almost identical, apart from an vertical offset.

be referred to respectively as *L0.5* and *L1.0*. *L1.0* appears to be similar to that of *L0.5*, but with a negative offset downwards. Closer examination however reveals that while the two curves are very similar, they are not identical. The cross-correlation coefficient between the two curves was 0.98. This suggests a very strong positive correlation.

The λ value thus functions very much like the threshold value th_{peak}. It has the effect of shifting the ΔBIC curve up or down. The further effect of λ on the segmentation $Recall$, $Precision$ and $FScore$ is shown in Fig. 3.6.

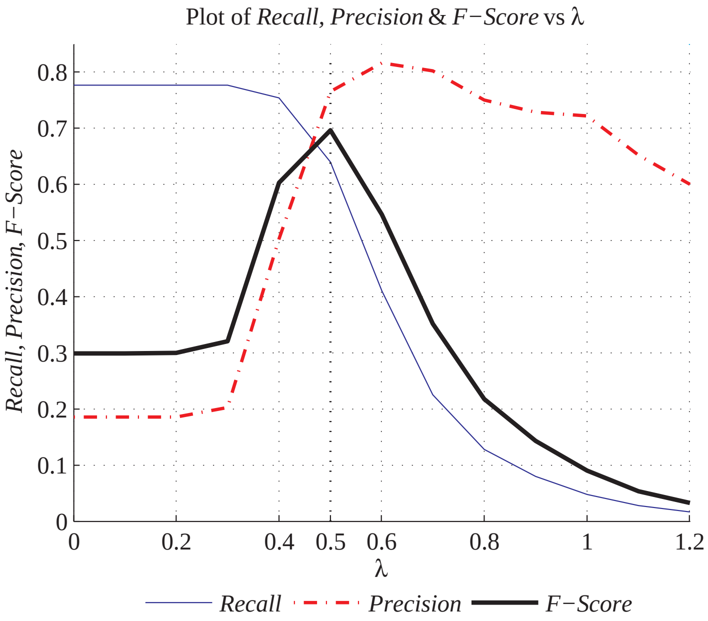


Figure 3.6*: The* $Recall$*,* $Precision$ *and* $FScore$ *obtained obtained for the BIC system as* λ *is varied. Results obtained using the Hub4-97e corpus.*

It can be seen that the variation of λ leads to a trade-off between the $Recall$ and $Precision$. A small λ leads to a very good $Recall$ but poor $Precision$. This can be explained by the fact that a small λ results in a ΔBIC curve that is higher. As such, the peak detection algorithm detailed in Section 3.2.3 will detect more peaks, resulting in a higher $Recall$. A large portion of those peaks detected however will be spurious. This thus explains the lower $Precision$.

As is shown in Fig. 3.6, The best λ value for the Hub4-97e corpus was found to be $\lambda = 0.5$. The $FScore$ at that point was found to be 0.697. The parameter configurations for the experiment is tabulated in Table 3.2.

Table 3.2: Parameters resulting from experiment to find the optimum λ for the BIC system

(a.) Parameters:	
Segmentation system :	*BIC*
optimal λ :	0.5
th_{peak} :	0
Δt_{collar} used :	0.5 second

(b.) Results:	
Recall :	0.639
Precision :	0.766
FScore :	0.697
# of turns detected :	462
# of turns deemed correct :	354
# of deletion errors :	200
# of insertion errors :	108

3.2.5 Segmentation using Hotelling's T^2 statistic

The T^2 equation used for the subsequent experiments is that from Eqn. 2.32.

$$T^2(\boldsymbol{\Delta}) \;=\; \frac{N_{\mathcal{L}}N_{\mathcal{R}}}{N_{\mathcal{L}}+N_{\mathcal{R}}}(\boldsymbol{\mu}_{\mathcal{L}}-\boldsymbol{\mu}_{\mathcal{R}})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_{\mathcal{L}}-\boldsymbol{\mu}_{\mathcal{R}}) \qquad \text{(Eqn. 3.2)}$$

These experiments are meant to reveal the differences between the T^2 and *BIC* algorithms, as well as to explore if a fusion of T^2 and *BIC* yields better results than using either singularly. Good results have previously been reported in [Zhou & Hansen, 2005] for a fusion of the T^2 and *BIC* algorithms. The experiments will be carried out as follows.

(i) Finding the optimal th_{peak} for T^2.

(ii) Examining the results from the fusion of T^2 with *BIC*.

To allow for a fair comparison, the feature vectors and windowing algorithm used are similar to that used in Section 3.2.4 for the *BIC* experiments. Unlike the *BIC* algorithm, the T^2 does not have an inherent performance tuning penalty factor λ. The trade-off between the *Recall* and *Precision* will thus be explored using the peak detection threshold th_{peak}.

3.2.5.1 The optimal peak detection threshold (th_{peak}) for T^2

Fig. 3.7 shows the T^2 curve against that of the ΔBIC for the same segment. It can be seen that the T^2 curve is somewhat less accurate and more spurious turn points are found. As will be shown next, this thus translates to a higher turn point sensitivity (higher $Recall$), at the expense of lower $Precision$.

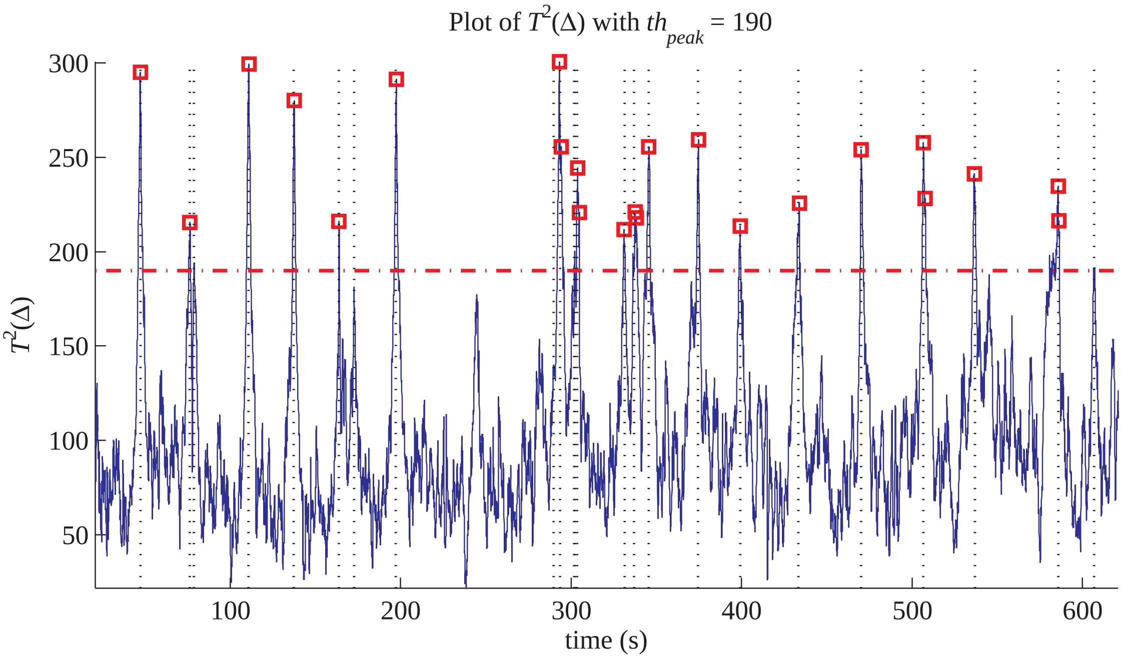
Unlike the computation of ΔBIC, the $T^2(\boldsymbol{\Delta})$ does not have an inherent penalty factor that can be adjusted. The peak detection threshold th_{peak} is thus used instead for the $T^2(\boldsymbol{\Delta})$ curve. The peak detection algorithm (See Section 3.2.3) that will be used in this section is the same as that used for *BIC*. th_{peak} will be adjusted to determine its effect on peak detection performance. A lower th_{peak} can be used to detect more turns points at the expense of $Precision$ while a higher th_{peak} will improve $Precision$ but reduce $Recall$. A $Recall$ versus $Precision$ trade-off point can thus be found where the algorithm's performance as measured by $FScore$ is at its optimum.

Fig. 3.8 shows the $Recall$ versus $Precision$ trade-off curves for the T^2 algorithm. The windowing algorithm used in this experiment is the same as that used for the *BIC*. The results showed that the best $FScore$ performance was worst than the *BIC*.

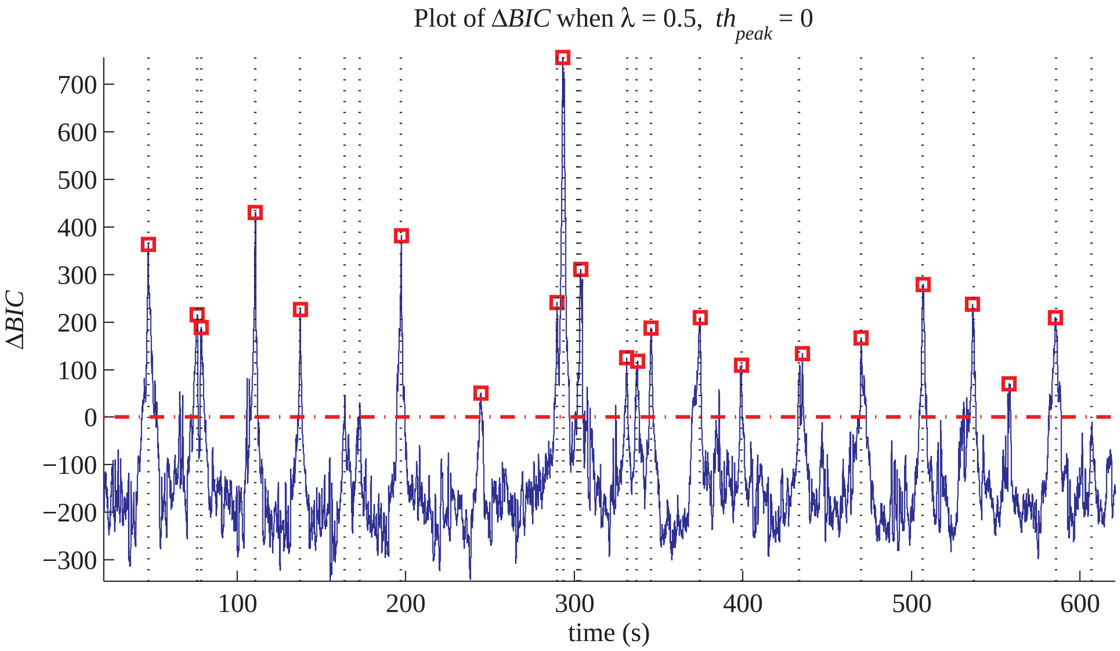
The T^2 method however was capable of yielding a maximum $Recall$ of 0.83 at a th_{peak} threshold of 60. This fares better than the maximum $Recall$ obtained by the previous *BIC* experiment. The higher $Recall$ for the T^2 method was however achieved with a very poor $Precision$ of 0.0851. Nevertheless, the high $Recall$ suggests a potential use for the T^2 as a compliment to the *BIC*, as was done in [Zhou & Hansen, 2005] and will be explored in the next section.

3.2.5.2 Fusion of T^2 and BIC

In view of the high $Recall$ potential of the T^2 algorithm, an experiment was thus done to determine the merits of fusing T^2 and *BIC*. A 2-stage fusion strategy similar to that



(a.)



(b.)

Figure 3.7: For a segment of audio from Hub4-97e, time series plot of (a.) T^2 using $th_{peak} = 190$ (b.) ΔBIC using $\lambda = 0.5$, $th_{peak} = 0$. Vertical lines indicate actual speaker turns. Square markers (□) indicate speaker turns detected.

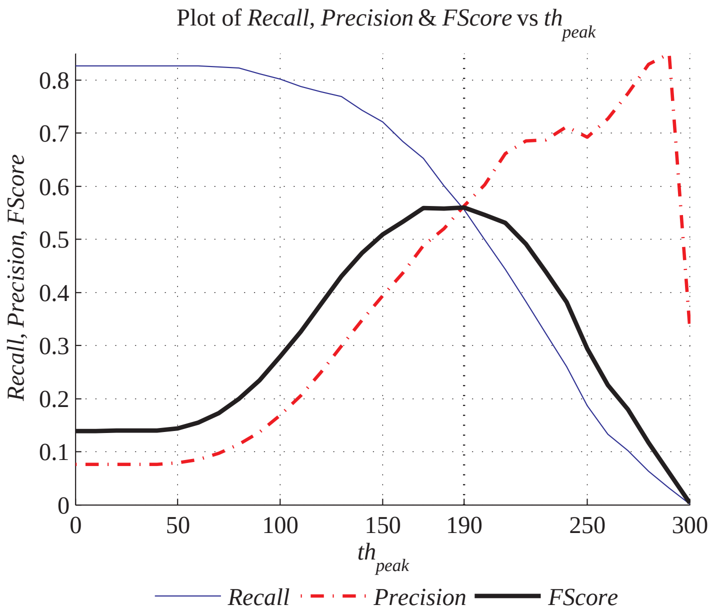


Figure 3.8: The Recall, Precision and FScore obtained for the T^2 system as th_{peak} is varied. Results obtained using the Hub4-97e corpus.

in [Zhou & Hansen, 2005] was used. The first stage involved using T^2 to identify a set of all probable turn points. This is where the relatively faster speed and higher *Recall* of the T^2 algorithm is used to perform a quick selection of likely turn locations. Portions of the corpus with low potential of containing turn points can thus be skipped over in the second stage of testing.

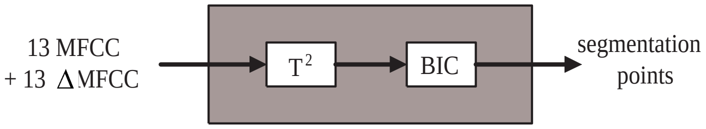


Figure 3.9: Block diagram of the T^2 and BIC fusion.

In the second stage, each of those points identified in the first stage are then evaluated using the *BIC* algorithm. By restricting the *BIC* testing to only locations short-listed in the first stage, unnecessary computation is avoided and the relatively slower *BIC* segmentation stage is sped up. The relatively higher *Precision* of the *BIC* algorithm is important here since it will result in a more stringent turn point selection than using the T^2 alone, effectively reducing the number of False Alarm turns.

In our experiment, we have chosen a value of $th_{peak} = 90$ as the *Recall* at this point is 0.802 and is not far off from the maximum *Recall* of 0.826 obtained at $th_{peak} = 60$. At this setting, only 2568 points have to be tested in the second stage, compared to the 5253 if the second stage was performed at $th_{peak} = 60$.

The ΔBIC testing of the hypothesized turn points from the first stage uses the following algorithm:

```
Windowing algorithm for the BIC stage of the T$^2$+BIC system:

  (i)  Initialize $\mathcal{BIC} = \{\varnothing\}$, $pRecent = \varnothing$

  (ii) For each point $p1$ in $\mathcal{T2}$

         (a)    Let
                   $t_{p1}$ = time at point $p1$
                   $t_{pRecent}$ = time at point $pRecent$
                   $t_{start} = \max\,[t_{pRecent}, t_{minimumLeft}]$ seconds
                   $i = 1, t_{end} = t_{p1+i}$ seconds
                   $b[t_{p1}] = 0$

         (b)    do
                   $b[t_{p1}] = \max\,[b[t_{p1}], \Delta BIC(t_{start}, t_{p1}, t_{end})]$
                   $i = i + 1, t_{end} = t_{p1+i}$ seconds
                repeat while $t_{end} - t_{p1} < \max\,\{t_{p1} - t_{start}, t_{maximumRight}\}$

         (c)    If $b[t_{p1}] > th_{peak2}$ then
                   $\mathcal{BIC} = \{\mathcal{BIC} \cup p1\}$
                   $pRecent = p1$
                end

       end
```

The second stage of the experiment takes in the set of potential turn points from the first stage and performs a ΔBIC test at every point. Let $\mathcal{T2}$ represent the set of all turn points detected by the T^2 algorithm of the first stage. $\mathcal{BIC}$ will represent the set of all turn points detected by the *BIC* algorithm in the second stage.

The value of λ in the ΔBIC equation (See Eqn. 3.1) once again is used as the performance tuning parameter. It is varied in order to adjust the trade-off between the *Recall* and *Precision* of the system. This trade-off curve is plotted in Fig. 3.10. It can be seen that an optimal value of λ would be 0.54 and the best F-Score was 0.622.

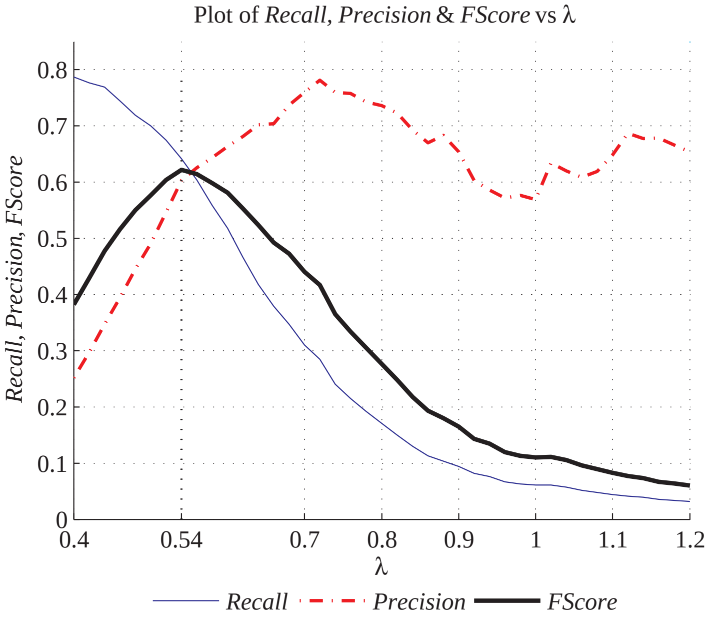


Figure 3.10: *The Recall, Precision and FScore of the combined* T^2*+BIC system as* λ *is varied.*

3.3 Discussions about the Speaker Segmentation Experiments

Table 3.3 reproduces the best segmentation performances of the systems examined in the earlier experiments. It can be seen that the *BIC* produced the best overall *FScore*. This result is better than the 0.637 reported in [Anguera, 2005] using *XBIC*, while being comparable to the *FScore* of 0.67 reported in [Ajmera *et al.*, 2004] for a *BIC*-based system. There however are systems such as [Chen & Gopalakrishnan, 1998b; Tritschler

& Gopinath, 1999; Liu & Kubala, 1999; Zhou & Hansen, 2000; Vandecatseye & Martens, 2003; Wu *et al.*, 2003] which reported better $FScore$ results (See Table 3.1). At the optimal functioning point of *BIC*, the $Recall$ and $Precision$ were somewhat balanced at 0.639 & 0.766 respectively.

Table 3.3: Optimal segmentation performance for the BIC, T^2 *& T*^{2}*+BIC algorithms. This table compares the best performance obtained from each separate system. The time taken for each algorithm is also listed.*

(a.) Common parameters:

Δt_{collar} used : 0.5 second

(b.) Results:

System:	*BIC* [1] $th_{peak} = 0$ $\lambda = 0.5$	T^2 [2] $th_{peak} = 190$	T^2 [3] $th_{peak} = 90$	T^2+*BIC* [4] $th_{peak} = 90$ $\lambda = 0.54$ $th_{peak2} = 0$
$Recall$	0.639	0.556	0.802	0.641
$Precision$	0.766	0.563	0.169	0.604
$FScore$	0.697	0.560	0.279	0.622
FAR (%)	23.4	43.7	83.1	39.6
MDR (%)	36.1	44.4	19.8	35.9
# of turns detected	462	547	2627	588
# of turns deemed correct	308	301	444	355
# of deletion errors	200	246	110	199
# of insertion errors	108	239	2183	233
Computational time required (minutes)	32	20	21	23
Computational factor (× real time)	0.18	0.11	0.12	0.13

[1] Optimal performance for *BIC* only system.
[2] Optimal performance for T^2 only system.
[3] T^2 first stage for the optimal T^2+*BIC* performance.
[4] Optimal performance for T^2+*BIC* two-staged system.

The best performance delivered by the T^2-based system was worst than the same for *BIC*. At its best, the T^2 detected 547 turns while the same for *BIC* detected only 462. While the T^2 detected more turns, its $Recall$ was still lower than that for *BIC*.

It was discussed earlier in Section 3.2.5.1 that by sacrificing the $Precision$ and $FScore$, it was possible for the T^2 to yield a higher maximum $Recall$ than that for *BIC*. At the point where the th_{peak} for T^2 was 90, the $Recall$ was 0.802. This is still higher than the maximum $Recall$ of 0.77 at $\lambda = 0.2$ achievable for the *BIC*. The value of $th_{peak} = 90$ was thus adopted for a fusion system combining the T^2 with the *BIC*. A caveat to note about using the *Recall* of 0.802 for the first step is that this would serve as an upper bound for the *Recall* of the fused system. In other words, turn points that missed shortlisting by the T^2 would not be considered for testing in the second stage.

The fusion experiment carried out in Section 3.2.5.2 unfortunately did not yield results comparable to that reported in [Zhou & Hansen, 2005]. In that paper, a similar T^2+*BIC* fusion was pursued and a *FScore* of 0.803 was obtained. It was also reported in that paper that the fused system could yield a better *FScore* than a system that used only *BIC*. Unlike what was reported in the paper, the fused system implemented in Section 3.2.5.2 did not yield better results than the *BIC* only system. The possible reason for the difference between the experimental results and what is reported in [Zhou & Hansen, 2005] would probably lie in the windowing algorithm used for the second stage. [Zhou & Hansen, 2005] does not report in detail the windowing algorithm they used when integrating *BIC* together with the T^2 front-end.

The computation time required for each of the segmentation methods is also listed in Table 3.3. It can be seen that while the T^2+*BIC* algorithm is faster than the *BIC*, the time savings was not on the order of 100× as cited in [Zhou & Hansen, 2005]. The system using T^2+*BIC* was observed to be about 1.4× faster than a comparable system using *BIC*. The time measurements were all taken off Matlab experiments conducted upon a 2GHz Pentium IV computer.

3.4 Clustering Results Reported on the Hub4-97e

The clustering problem addresses the issue of grouping the segments found into individual speaker class. The following are some clustering results that have been reported upon the Hub4-97e database.

Table 3.4: Clustering results reported on the Hub4-97e database.

System	# clusters at end point, $N_{clusters}$	ACP (%)	Average number Speakers Per Cluster, $ANSpC$
[Harris *et al.*, 1999]			
KL distance	not stated	89.10	non stated
[Tritschler & Gopinath, 1999]			
ΔBIC "offline"	172	96.70	non stated
ΔBIC "online"	149	98.58	non stated
[Zhou & Hansen, 2000]			
ΔBIC	non stated	96.30	1.13
Gender classification + BIC	not stated	99.30	1.06
[Vandecatseye & Martens, 2003]			
normalized ΔBIC	160	89.00	non stated

In comparing the clustering performance across different systems, it is important to be mindful that the resultant ACP is dependent on the number of clusters at the stopping point ($N_{clusters}$). A system that terminates clustering earlier (higher $N_{clusters}$) typically will have a better ACP than a similar system where clustering is terminated later. For the Hub4-97e corpus, there should ideally be 117 clusters (speakers) upon termination. Most systems will terminate clustering before 117 clusters are obtained (i.e. to under-cluster). This is to prevent the resultant cluster ACP from becoming too poor. In the papers [Tritschler & Gopinath, 1999; Zhou & Hansen, 2000], the clustering is reported to terminate at 149 (for "online" clustering) and 160 clusters respectively. High ACP of 98.58% and 89.00% were correspondingly obtained.

[Zhou & Hansen, 2005] also employed the BIC in their clustering algorithm. They however introduced an improvement by performing clustering as a 2 step process. The first step would be to classify each segment according to its gender. Gender classification is performed using a pair of GMMs - one for male and another for female. Good gender classification accuracy was reported. 96.4% of male speech segments were correctly classified while the same for female segments was 99.1%. The second step would then be to carry out hierarchical clustering using the BIC within each gender pool. The idea underlying this two step approach is that since gender classification can be performed to a very high level of accuracy, this will thus have a positive knock-on effect for subsequent speaker clustering. ACP of 99.30% was reported along with a $ANSpC$ of 1.06.

3.5 Summary of this chapter

This chapter evaluated the BIC and T^2 divergence measure when performing segmentation on the Hub4-97 broadcast news corpus. The BIC divergence measure was first evaluated with the objective of repeating the results obtained in [Ajmera *et al.*, 2004]. The results we obtained were close to that reported in the paper.

An experiment was then performed to evaluate segmentation where T^2 is used as a first pass before BIC (i.e. the $T^2 + BIC$ system as was reported in [Zhou & Hansen, 2005]). The results obtained for this experiment were unfortunately not as successful and did not prove to be better than that for the BIC only system. The experiment however did show that a two-stage system had the potential to perform faster than using just the BIC.

All experiments conducted so far have been done on the broadcast news domain and the following chapter will now focus on performing diarization on meeting recordings. This would be a scenario quite different from that of broadcast news recordings because multiple recording channels will now be available and the speaking characteristics of people in the meeting environment will differ from that in broadcast news. A system for performing automatic speaker diarization will be proposed, along with results obtained using the RT-06s and RT-07 audio corpora.

Chapter 4

Speaker Diarization of Meeting Recordings

4.1 Speaker Diarization for Meeting recordings

In this chapter, we will examine the speaker diarization task for meeting recordings. Specifically, we will use the US National Institute for Science and Technology (NIST) Rich Transcription (RT) diarization evaluations for meeting proceedings recorded across multiple distant microphones (MDM).

Given the nature of meeting recordings, the speaker diarization of meetings requires processing steps beyond the speaker segmentation and clustering that is performed for broadcast news. The audio recordings are less controlled and contain more non-speech events such as breathing noises, coughs or lip-smacks. It also contains longer segments of silence where nobody speaks. The issue of overlapping speech also represents another challenge. The meeting recordings domain as compared to broadcast news however has the benefit of having multiple recording sensors. This thus provides a diversity of perspectives from the different channel recordings.

Much of the recent research on speaker diarization of meetings concentrates on exploiting the differences in the diversity of microphone recordings. These approaches can be generalized into 3 categories - generating an enhanced channel out of the multitude of channels [Istrate *et al.*, 2005; Anguera *et al.*, 2005a; Fredouille & Evans, 2007], performing diarization on each channel and then fusing the diarization results [Meignier *et al.*, 2000; Moraru *et al.*, 2003; Fredouille *et al.*, 2004; Fredouille & Senay, 2006], or using the multitude of channels to perform some form of speaker localization using the time delays between channels [Ellis & Liu, 2004; Wooters *et al.*, 2004; Anguera *et al.*, 2005b; Wooters & Huijbregts, 2007; Koh *et al.*, 2007a].

This chapter is organized as follows. It begins with Section 4.1 giving an introduction to existing speaker diarization systems employed in the meeting room domain. This is followed by Section 4.2 which describes the NIST Rich Transcription audio corpus and evaluation rules. Section 4.3 then describes a system developed for participation in the NIST RT-07 evaluation. The diarization performance of this system on the NIST RT-06s and RT-07 corpora are then examined in Section 4.4 and Section 4.5.1 then concludes with a discussion of some issues affecting the quality of speaker diarizations.

4.2 The NIST RT Evaluation Environment

Since 2004, the National Institute for Standards and Technology (NIST) has been conducting yearly evaluations for meeting room speaker diarization systems. These evaluations are held under the Rich Transcription (RT) [NIST, 2007; NIST, 2006c] framework. The RT evaluations have since become the defacto platform upon which teams would evaluate and report their speaker diarization system.

The NIST RT evaluation corpora for each year typically consists of about 3 hours of meeting recordings (Table 4.1). This is divided into 8 parts, each part the recording is that of a different meeting. The recordings would have been made in various meeting rooms. The meeting rooms vary in their sizes, microphone positions, microphones used*,

*Data is collected using multiple distant microphones but the model of the microphones used for the different meeting rooms is not revealed.

as well as the general acoustic properties. This would thus mean that parameters often have to be tailored for individual meeting rooms and some meeting rooms tend to yield better diarization results than others.

Table 4.1: Characteristics of the tasks in the RT-06s & RT-07 evaluations.

Task name	duration (minutes)	# channels available	possible pair permutations	meeting topic
RT-06s tasks				
CMU_20050912-0900	17.8	2	2	Transcription team meeting
CMU_20050914-0900	18.0	2	2	Transcription team meeting
EDI_20050216-1051	18.0	16	240	Remote control design
EDI_20050218-0900	18.2	16	240	Remote control design
NIST_20051024-0930	18.1	7	42	Project planning meeting
NIST_20051102-1323	18.2	7	42	Data resource planning
VT_20050623-1400	18.0	4	12	Problem solving scenario
VT_20051027-1400	17.7	4	12	Candidate selection
RT-07 tasks				
CMU_20061115-1030	22.5	3	6	Discussion group
CMU_20061115-1530	22.6	3	6	Transcription team meeting
EDI_20061113-1500	22.6	16	240	Remote control design
EDI_20061114-1500	22.7	16	240	Remote control design
NIST_20051104-1515	22.4	7	42	Planning meeting
NIST_20060216-1347	22.5	7	42	SWOT analysis meeting
VT_20050408-1500	22.4	4	12	Problem solving scenario
VT_20050425-1000	22.6	7	42	Problem solving scenario

The rules for the NIST RT evaluations have been consistent since 2004. The goal of each speaker diarization system is to answer the question of “Who Spoke When ?”. This consists of indicating the start and end times for every utterance in the recording and attributing each utterance to a speaker identity. Background segments where nobody speaks do not have to be explicitly indicated. Non-speech sounds should be regarded as background segments and excluded from the diarization. These non-speech sounds can be vocalized sounds such as laughter, coughs, sneezes and breathing noises, or environmental sounds like knocks and claps. A 0.5 seconds (± 0.25 seconds around the ground truth) forgiveness collar is applied to each start and stop time stamp.

4.2.1 Evaluation measures for speaker diarization in meetings

Diarization systems performing speaker diarization upon the NIST RT evaluation corpora will usually report their performance using the Diarization Error Rate (DER). It essentially is the ratio of all diarization times that are in error, against the sum total of all speaker time. As such, the perfect DER score will be 0% while 100% would be the worst possible result. Fig. 4.1 illustrates the various types of errors considered in the DER equation.

$$\begin{aligned} DER &= \frac{\text{sum of all diarization time that is erroneous}}{\text{sum of all speaker time}} && \text{(Eqn. 4.1)} \\ &= \frac{SE + MS + FA}{SPK}(\%) && \text{(Eqn. 4.2)} \end{aligned}$$

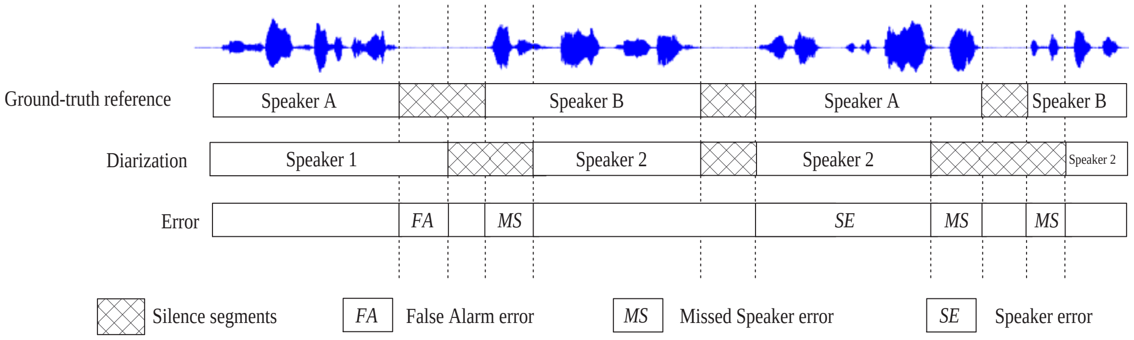


Figure 4.1: The time components that contribute to the DER.

- Speaker Error time (SE): Total time that is attributed to the wrong speaker. This refers to speech segments belonging to an arbitrary Speaker A that has been assigned incorrectly to Speaker B.

- Missed Speaker time (MS): Total time in which less speakers are detected than what is correct. An example of such an error is where there are two speakers talking simultaneously, but only one speaker is detected in the meeting.

- False Alarm Speaker time (FA): Total time in which more speakers are detected than what is correct. An example of such an error is where there is only one speaker talking but multiple speakers are erroneously detected, or where no one is talking but the system detects at least one speaker.

- Scored Speaker time (SPK): Sum of every speaker's utterance time as indicated in the reference. The SPK can be longer than the duration of the entire corpus because segments containing multiple overlapping speakers will be counted once for each speaker.

Meeting recordings tend to contain segments of overlapping speech. As such, the DER reported for a meeting diarization system will thus usually define as to whether it included overlapping speech in the scoring. In systems that have not been designed to cope with overlapping speech, results are sometimes reported excluding those overlapping segments.

Another commonly reported diarization result would be the "Speaker as Speech Activity Detection" DER, or $SAD\ DER$. This performance measure reports the Speech Activity Detection (SAD) performance of the system. The task underlying SAD would be to accurately determine all the time spans where at least one or more speakers are speaking. Time instances were no one is speaking should also be correctly indicated as such. The $SAD\ DER$ is not concerned about the identity of the speakers present, nor about the presence of overlapping speakers. Where more than one speaker is speaking at a time instance, the SAD result only needs to indicate that a speaker is speaking. It does not need to specify the number of concurrent speakers, or the speaker identity.

The $SAD\ DER$ can thus be defined to be

$$SAD\ DER = \frac{MS_s + FA_s}{SPK_s}(\%) \qquad \text{(Eqn. 4.3)}$$

MS_s, FA_s and SPK_s respectively denote the amount of missed speaker time, false alarm speaker time and total scored speaker time, ignoring the presence of overlapping speakers. Unlike the previous definitions of MS, FA and SPK, segments containing multiple speakers will only be counted once. A lower $SAD\ DER$ would thus suggest better SAD performance and $SAD\ DER = 0\%$ would mean the perfect detection of all time instances where someone is speaking.

4.2.2 Results reported for NIST RT-06s & RT-07

Table 4.2 and 4.3 shows some of the best diarization results reported on the NIST RT-06s and RT-07 evaluation corpora.

Table 4.2: List of speaker diarization results reported on the NIST RT-06s evaluation corpora

System	official reference		word-aligned reference †
System	*DER* (%)	*SAD DER* (%)	*DER* (%)
[Rentzeperis *et al.*, 2006]			
AIT	70.70	11.00	-
[Leeuwen & Huijbregts, 2006]			
AMI	44.80	4.30	-
[Fredouille & Senay, 2006]			
LIA	38.80	4.70	-
[Janin *et al.*, 2006]			
ICSI	35.80	23.50	21.19
[Koh *et al.*, 2007b]			
I^{2}R/NTU	31.02	6.65	25.83

†: These results were obtained using the word-aligned references that [Anguera, 2006a] produced using the ICSI-SRI speech recognition system.

Some of the results for the NIST RT-06s evaluation were scored against a revised word forced-aligned reference generated by the ICSI [Anguera, 2006a]. This reference was generated using the ICSI-SRI Speech-to-Text recognition system [Stolcke *et al.*, 2005]. The time stamps used in the diarization scoring are aligned to the time boundaries of the words resulting from the speech recognition system. The reason for the revised reference is to address issues regarding the lack of consistency in human annotated references. This variability of human transcriptions is not an issue for results reported on the NIST RT-07 corpus because the official references released by NIST by default had word-alignment applied.

Table 4.3: List of speaker diarization results reported on the NIST RT-07 evaluation corpora

System	official reference		Average # speakers ‡	% tasks with correct number speakers ‡
	DER (%)	*SAD DER* (%)		
[Zhu *et al.*, 2007]				
LIMSI	26.07	3.23	12.3	12.5
[Fredouille & Evans, 2007]				
LIA	24.16	3.69	4.9	12.5
[Luque *et al.*, 2007]				
UPC	22.70	5.39	3.9	25.0
[Leeuwen & Konecny, 2007]				
AMIDA	22.03	6.73	7.1	0.0
[Koh *et al.*, 2007a]				
I^{2}R/NTU	15.32	8.65	4.4	75.0
[Wooters & Huijbregts, 2007]				
ICSI	8.51	3.33	4.5	87.5

‡: These results were obtained from [Fiscus *et al.*, 2007]

4.3 Proposed Speaker Diarization System for Meeting Room Recordings

This section discusses in detail the speaker diarization system submitted by the I^{2}R & NTU* for the NIST RT-07 evaluation. The system was submitted for benchmarking in the NIST RT-07 evaluation exercise and obtained an overall second placing. This work also resulted in two conference papers by the author and the I^{2}R & NTU team. Said papers are listed in Appendix A.

The diarization system can be divided into 4 main modules:

*The system described is a joint effort between the Institute for Infocomm Research (I^{2}R) and Nanyang Technological University (NTU). The author is a member of the team that developed said system.

(i) Segmentation using TDOA estimation

(ii) Bootstrap clustering of TDOA estimation

(iii) Speaker cluster purification

(iv) Non-speech & silence removal

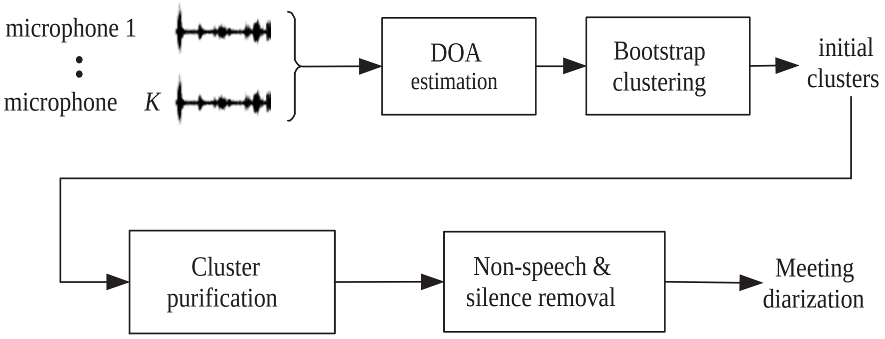


Figure 4.2: The block diagram of the proposed speaker diarization system.

This system was designed to utilize the diversity of the multiple directional microphone recordings in two ways - by performing speaker localizations using the Time Delay of Arrival (TDOA) estimates and by using delay-and-sum beamforming to produce an enhanced signal channel. The TDOA estimates across the various microphone pair permutations is first computed and a histogram-based quantization technique is then used to perform segmentation and bootstrap clustering. The initial speaker clusters from the bootstrap clustering process are then subjected to a purification process. In this step, the feature vectors used are extracted from the enhanced channel. The clusters are purified using an iterative GMM adaption and classification process. This process is repeated until such a point when the cluster assignments have stabilized. The final diarization step would then be to perform non-speech & silence removal. This is done to reduce the amount of false alarm (FA) speaker time in the diarization.

The following sections will now elaborate on each module to a greater detail.

4.3.1 Time Delay of Arrival (TDOA) Estimation using a NLMS filter

The time delay of arrival (TDOA) of speech between each microphone pair is estimated using a Normalized Least-Means Squared (NLMS) filter [Haykin, 2001]. Given an arbitrary microphone pair, this involves designating one audio channel as the source $\mathbf{s}[t]$, and the other as the reference $\mathbf{r}[t]$. The purpose of the filter is to converge the source signal with the reference by way of a stochastic gradient descent algorithm. The filter coefficients are continuously being updated whenever the signal Teager energy [Kaiser, 1990] of the reference channel is deemed to be sufficiently high. An estimate of the Time Delay of Arrival (TDOA) can then be obtained by seeking the highest weight in the filter coefficients.

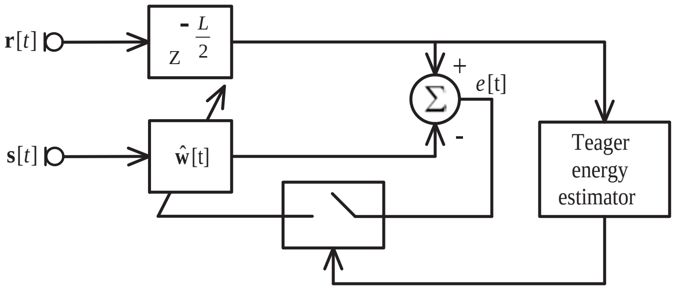


Figure 4.3: The block diagram of the NLMS filter used to perform TDOA estimation.

The filter adaptation is performed by sliding a L-sample long sliding window across two audio channels of the corpus. The L samples within the window will constitute a frame. The window is shifted by L samples in each step and this shifting is done in unison for both the source and reference channels. There are thus no window overlaps between consecutive frames. The filter coefficients at an instance of the *NIST_20051104-1515* corpus is shown in Fig. 4.4. A filter length of $L = 250$ was used for that corpus. The TDOA estimate at that time instance will be the index of the peak coefficient.

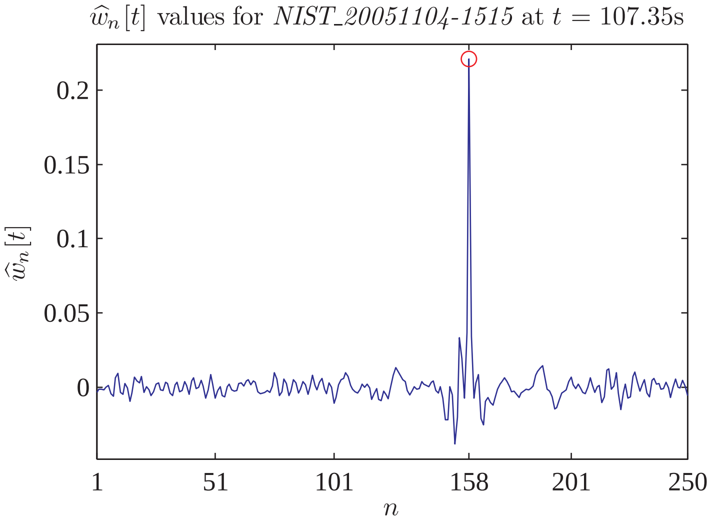


Figure 4.4: $\hat{\mathbf{w}}[t] = [\hat{w}_0[t] \cdots \hat{w}_n[t] \cdots \hat{w}_{L-1}[t]]^T$ *values where a single peak is observed at* $n =$ *158.*

A single peak can be observed at the 158th coefficient. Given that $L = 250$, the delay between the two channels can be found to be $158 - \frac{L}{2} = 28$ samples. A sampling rate (f_s) of 16000 samples per second is used in the recording. This thus translate to a time delay of 1.75 ms.

For a recording consisting of $N_{recording}$ frames and having K different microphone pair permutations, the TDOA across time can be represented as

$$\mathbf{TDOA}[t,k] = \underset{j=1..L}{\arg\max}\{w_j[t,k]\} \qquad \text{(Eqn. 4.4)}$$

where $t = 1..N_{recording}$ and $k = 1..K$. $w_j[t,k]$ is the j^{th} filter coefficient for the k^{th} microphone pair at time t. A plot of the $\mathbf{TDOA}[t,k]$ across time t for an arbitrary k^{th} microphone pair is shown in Fig. 4.5. This is a plot for 200 seconds of the *CMU_20061115-1030* task. It can be observed that there are 4 horizontal “tracks” in the plot. Each “track” is denoted by a horizontal dotted line. The identity of the speaker talking at a particular time instance can thus be inferred from the “track” closest to the peak filter coefficient.

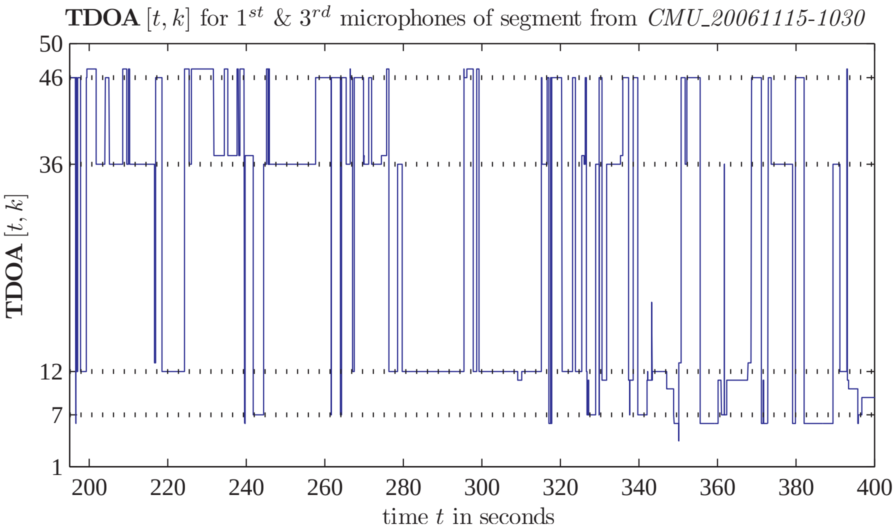


Figure 4.5: Plot of $\mathbf{TDOA}[t,k]$ *in a meeting where there are 4 speakers present.*

4.3.1.1 Choosing the NLMS filter length

The NLMS filter length L is chosen to ensure that L is capable of showing peaks at the largest extend of the TDOA dynamic range. Assume that we let $TDOA_{max}$ to be the maximum inter-channel delay present between all possible microphone pairs in a corpus. Using the filter setup that was described in the previous section, the TDOA range that is detectable can be represented centered around 0 as $[-TDOA_{max} \cdots 0 \cdots + TDOA_{max}]$. The value L can thus be selected to be

$$L = \frac{TDOA_{max}}{f_s} + tolerance \qquad \text{(Eqn. 4.5)}$$

where f_s is the sampling rate of the microphone recordings and $tolerance$ is some tolerance value to allow for estimation inaccuracies. The value of $TDOA_{max}$ unfortunately is not one that can be readily estimated by examining the signals. Another way of choosing L will be to select it using the maximum inter-microphone separation, $max\_separation$.

Where $separation_{i,j}$ is the distance separating the i^{th} microphone from the j^{th},

$$max\_separation = \max_{\forall i,j \in \{\text{set of all microphones}\}, i \neq j} \{separation_{i,j}\} \qquad \text{(Eqn. 4.6)}$$

For the RT-06s and RT-07 tasks, $L = 250$ was found to be a suitable value. Given that the sampling rate f_s is 16000 samples/sec, and assuming the speed of sound c to be 330m/sec, the length $L = 250$ is able to accommodate a maximum microphone pair separation of 2.7 metres.

4.3.1.2 Teager energy estimation

The audio recorded by each microphone will be corrupted by additive noise and the power of the speech signal $|\mathbf{s}[t]|^2$ will fluctuate throughout the meeting, depending on who and what is being spoken.

In the event that the power of $\mathbf{s}[t]$ is low, the noise energy will dominate in the recording. TDOA estimations at such instances are highly unreliable. Spurious TDOA values tend to occur at such instances and will result in poor TDOA based segmentation. One way to mitigate such a problem would be to estimate the energy of the signal while doing TDOA estimation. In the event that the signal energy is low, the TDOA is not estimated. Rather, the TDOA value for the present time t retains the previously estimated value, i.e.

$$\mathbf{TDOA}[t, k] = \mathbf{TDOA}[t-1, k] \qquad \text{(Eqn. 4.7)}$$

The energy of a signal at a time instance can be estimated using the Teager energy [Kaiser, 1990] as defined by

$$TE(\mathbf{r}[t]) = \mathbf{r}^2[t] - \mathbf{r}[t+1]\mathbf{r}[t-1] \qquad \text{(Eqn. 4.8)}$$

4.3.1.3 Microphone pair selection

In cases where many microphone recordings are available, the number of possible microphone pair permutations may be large. In recordings such as the *NIST_20051104-1515* where there are 7 distant microphones, the number of potential microphone pairs is $7^2 - 7 = 42$. Under such circumstances, we ought to judiciously choose pairs that exhibit characteristics typical of good TDOA estimation. The inclusion of microphone pairs with poor estimation capabilities will serve to deteriorate the TDOA estimation potential of the system.

The following three criteria were used to select microphone pairs for use:

(i) High Signal-to-noise ratio (SNR).

Given many microphones to choose from, the choice of those with the highest SNR is an obvious one. High SNR recordings allow better estimation of DOA from these signals.

(ii) Large average highest-peak to next-highest-peak ratio on $\mathbf{w}[t]$.

As will be elaborated in the subsequent Section 4.5.1, the coefficients of the NLMS filter can have multiple peaks. These peaks may correspond to the presence of signal reverberations, or the presence of frequency specific coloured noise. The presence of multiple peaks makes the estimation of TDOA difficult since these peaks will compete with each other to be the largest values. In the event that the wrong peak prevails, an incorrect TDOA estimate will be made.

This problem of competing peaks can thus be minimized by selecting only the microphone pairs which have a large average difference between the highest-peak and the next-highest-peak. A large difference between peaks will mean a greater margin for error and as such more accurate TDOA estimations.

(iii) Large TDOA dynamic range.

The dynamic range of the TDOA values has a direct relationship with the inter-microphone separation $separation_{i,j}$ & the azimuth θ (i.e. the angular orientation of the microphone). The TDOA values will have a large dynamic range when the microphones are positioned such that θ is small and $separation_{i,j}[t]$ is large [Varma, 2002]. A large dynamic range is desirable because it suggests that this will yield better resolution when performing bootstrap clustering.

Information about the positions of the microphones and human participants in each meeting unfortunately is not available for the RT-06s and RT-07 evaluations. The TDOA dynamic range of microphone pairs in such a scenario can thus be determined by doing actual TDOA estimation on the recordings. An excessively large filter length L can be used to perform exploratory TDOA estimations between microphone pairs. Given an arbitrary k^{th} microphone pair, the dynamic range will be $[min\_TDOA_k \cdots max\_TDOA_k]$ where

$$min\_TDOA_k = \underset{t=1..N_{recording}}{\arg\min} \quad TDOA[t,k] \qquad \text{(Eqn. 4.9)}$$

$$max\_TDOA_k = \underset{t=1..N_{recording}}{\arg\max} \quad TDOA[t,k] \qquad \text{(Eqn. 4.10)}$$

$N_{recording}$ is the length of the meeting recording. The microphone pairs with the top $\{max\_TDOA_k - min\_TDOA_k\}$ values can thus be chosen for further evaluation.

4.3.2 Bootstrap clustering using TDOA estimates *

Bootstrap clustering uses location information from Eqn. 4.4 to form initial clusters. This approach performs segmentation and clustering in a single joint step. This differs from the more traditional approach introduced in Chapter 3 where segmentation and

*It is to be noted that the bootstrap clustering module described in this section was developed by Dr. Sun Hanwu of the Institute for Infocomm Research (I^{2}R), Agency For Science, Technology And Research (A*STAR), Singapore.

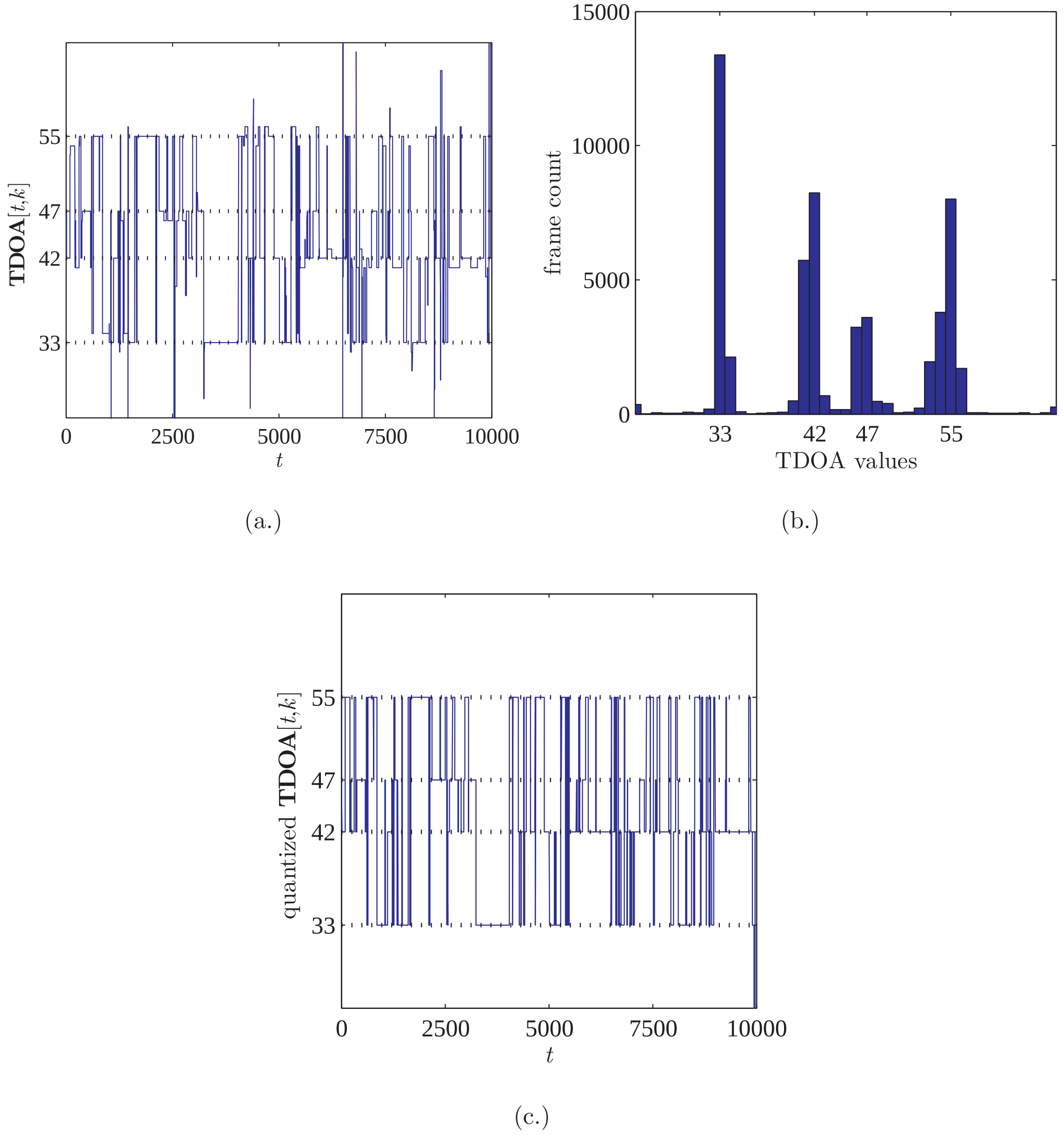
clustering are done separately. The method works by building histograms in a two step process. The first step is that of within-pair quantization of the TDOA estimates for each microphone pair (the columns of $\mathbf{TDOA}[t,k]$) to commonly occurring locations. The second step is that of inter-pair quantization to fuse information from all K microphone pairs (the rows of $\mathbf{TDOA}[t,k]$), forming a single decision about the likely speaker origins and thus effectively clustering the speakers. The number of unique clusters resulting basically equates to the number of speakers. These speaker clusters are then used as seeds in the cluster purification step.

It is important to note that the clustering that is performed in this stage forms speaker clusters using only spatial location information (i.e. speaker localization using TDOA estimates). As such, impure segments and clusters containing multiple different speakers will result if speakers move or change places during the meeting. This is one of the key shortcomings of this method and it will be discussed in a greater detail in Section 4.5.1.1. Iterative cluster purification using location independent acoustic features thus has to be performed subsequently in order to mitigate the presence of impurities in the post-quantization clusters.

4.3.2.1 Within-pair quantization

Within-pair quantization is applied to every combination of microphone pairs under consideration. The purpose of this is to quantize the TDOA estimates into a small set of commonly occurring values. In doing so, this will yield a set of segmentation and clustering hypotheses specific to the microphone pair.

Given an arbitrary k^{th} microphone pair consisting of two different channels, a histogram is first built using the $\mathbf{TDOA}[t,k], t = 1..N_{recording}$ values. $N_{recording}$ is the length of the meeting recording. Centroids are identified in the resulting histogram. These centroids refer to TDOA positions corresponding to peaks in the histogram. Every peak in



(a.)

(b.)

(c.)

Figure 4.6: Within-pair quantization. (a.) Plot of one column of $\mathbf{TDOA}[t,k]$. *Horizontal dotted lines correspond to histogram centroids. (b.) Histogram of TDOA values for selected* k^{th} *microphone pair. (c.)* $\mathbf{TDOA}[t,k]$ *values after within-pair quantization.*

the histogram indicates that there is a significant amount of speech originating from that particular location. An assumption is made that speech originating from a single location will very likely belong to a single homogenous speaker. As such, the number of peaks can be used as an estimate of the number of speakers present. Using a nearest-neighbour approach, all values in the histogram are thus quantized into their respective nearest centroids (Fig. 4.6).

$$th_{within} = \frac{N_{recording}}{N_{bins}} \qquad \text{(Eqn. 4.11)}$$

The decision as to what constitutes a peak is made using a threshold value th_{within}. Fig. 4.7 illustrates how the application of th_{within} generates "islands" in the histogram. Peaks are consequently selected to be the highest point within each "island". The number of "islands" detected would be the number of speakers deemed present in the recording.

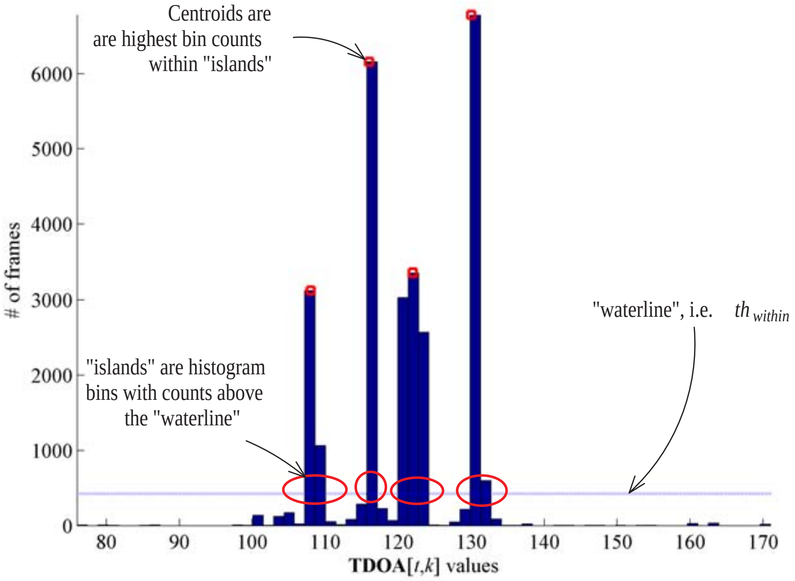


Figure 4.7: The application of a threshold or "waterline" forms "islands" of histogram bins that are above the threshold. Centroids are the highest bins within the "islands".

A fixed number of $N_{bins} = 40$ was used in the subsequent evaluations on the RT-06s and RT-07 corpora. This number was found to yield fairly accurate estimates of the number of speakers present when the actual number of speakers present is less than or equal to 6.

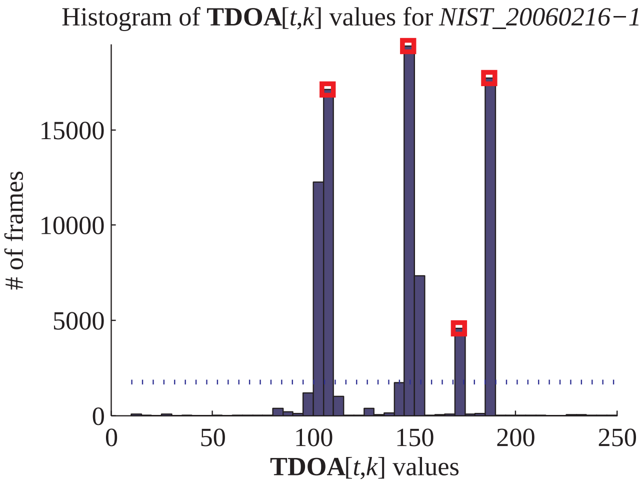


Figure 4.8: Histograms of $\mathbf{TDOA}[t,k]$ *for pair consisting of* 2^{nd} *&* 4^{th} *microphones from NIST_20060216-1347 . The histogram reflects 4 centroids when there are 6 unique speakers. A* N_{bins} *of 40 is used.*

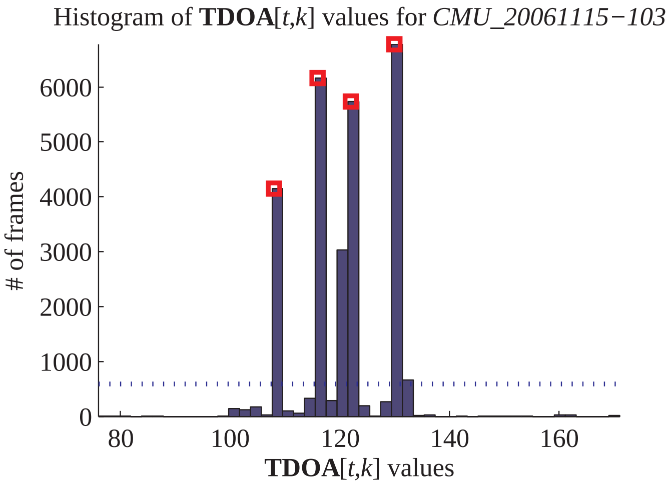


Figure 4.9: Histograms of $\mathbf{TDOA}[t,k]$ *for pair consisting of* 1^{st} *&* 2^{nd} *microphones from CMU_20061115-1030 . The histogram reflects 4 centroids corresponding to 4 unique speakers. A* N_{bins} *of 40 is used.*

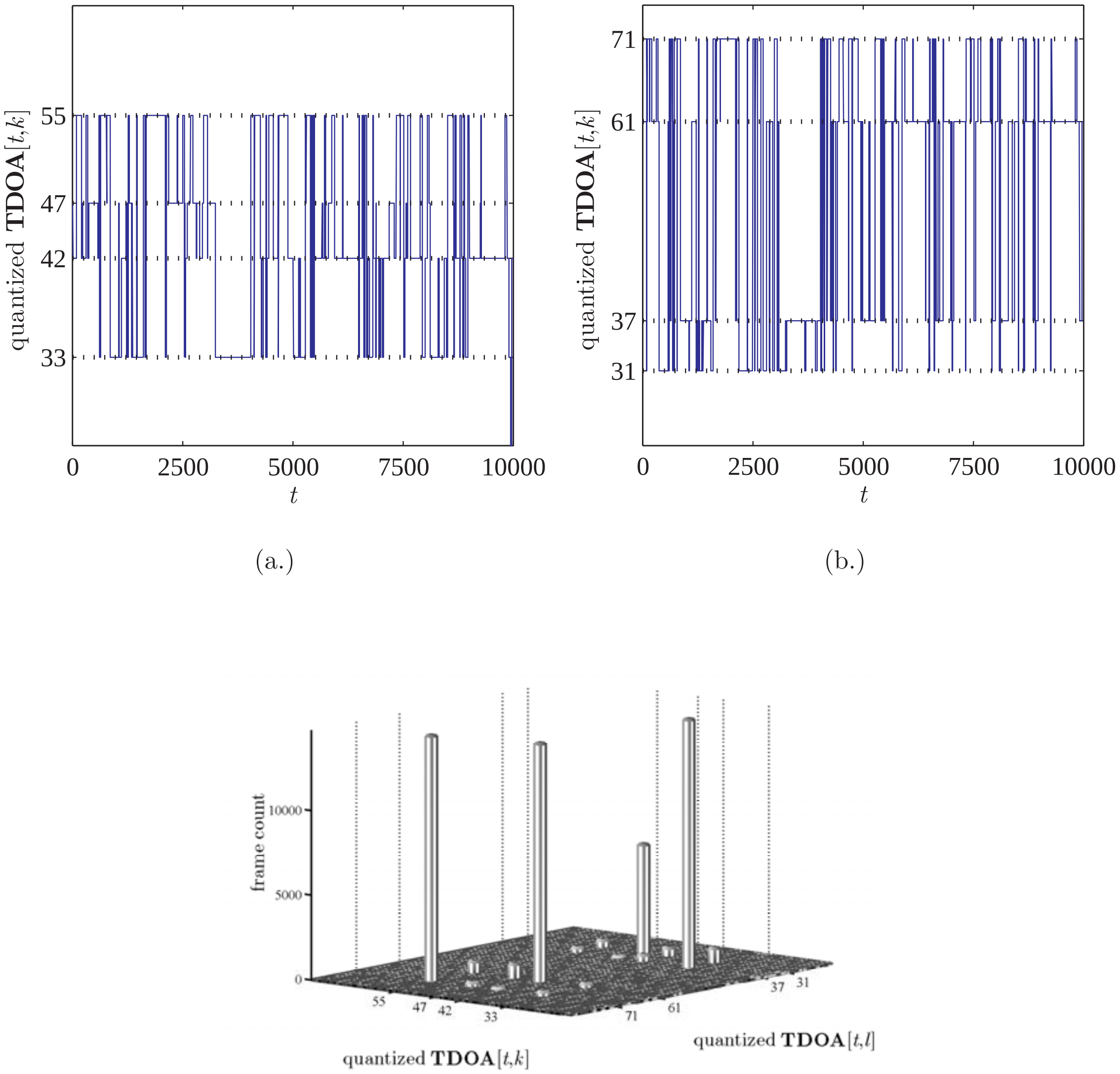
The results of performing quantization to the $\mathbf{TDOA}[t,k]$ can be seen in Fig. 4.8 & 4.9. In cases where the number of centroids misjudge the number of speakers, the subsequent inter-pair quantization step may still be able to recover from the error if the other microphone pairs for the task can give good estimations. In Fig. 4.8, 4 centroids are visible but there are 6 speakers. Fig. 4.9 shows a microphone pair yielding good quantization results. 4 clear and distinct centroids can be seen within the histogram.

By quantizing the $\mathbf{TDOA}[t,k]$ values (See Fig.4.6a) to the nearest centroid, a time-series plot (See Fig 4.6c) is obtained. This quantized value can then be passed on to the next inter-pair quantization stage.

4.3.2.2 Inter-pair quantization

The previous subsection discusses quantization along the columns of $\mathbf{TDOA}[t,k]$. The second step of this module is to perform quantization along the rows of $\mathbf{TDOA}[t,k]$, i.e., to identify centroids across K microphones pairs. Using the quantized results from within-pair quantization, a K-dimension histogram is built across all microphone pairs. Centroids can be readily identified within this high dimensional histogram. Like what is done in Within-pair Quantization, these centroids are selected by virtue of their relatively high bin counts. An illustration of this is shown in Fig. 4.10.c where 4 centroids can be observed by quantizing across 2 microphone pairs. The remaining histogram bins with low counts will then be clustered into the nearest centroid.

Experiments conducted on the RT-07 corpus showed that segments found after applying the bootstrap clustering were mostly of short durations - almost 90% of the segments are less than 3 seconds long and 71% of all the segments are shorter than 1 second. It is interesting to note that apart from *VT_20050408-1500* , the initial clusters resulting after this module were observed to yield reasonably low Speaker Error times SE (See Table 4.4). The subsequent step of cluster purification only serves to improve the DER



(a.)

(b.)

(c.)

Figure 4.10: For part of task CMU_20061115-1030 (a.) quantized **TDOA**$[t, k]$ *values for an arbitrary k^{th} microphone pair (b.) quantized* **TDOA**$[t, l]$ *values for an arbitrary l^{th} microphone pair (c.) 2-D histogram of quantized TDOA values for both k^{th} and l^{th} microphone pairs. 4 peaks are visible.*

by between 0% to 3.64% absolute. This thus shows the effectiveness of the clustering method when the TDOA estimations are accurate.

4.3.3 Further Processing*

4.3.3.1 Cluster purification using GMM

Clustering using only TDOA information will not be robust towards situations where the speakers move significantly during a meeting, or where the room is highly reverberant. In such situations, after bootstrap clustering, speech from a single speaker might span across multiple clusters. The resultant clusters will therefore be impure. Cluster purification using acoustic features will help to reduce erroneous clustering.

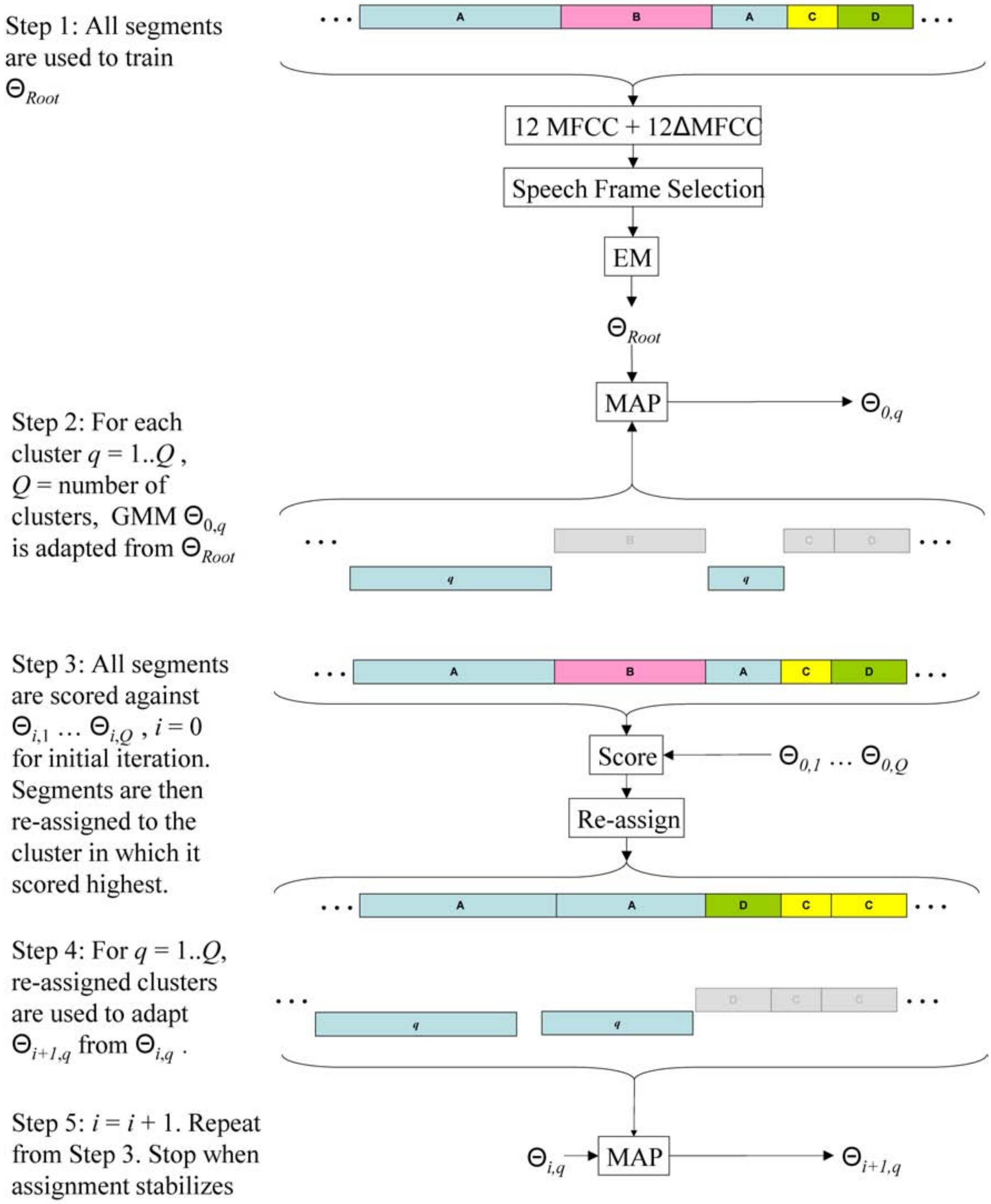
The cluster purification algorithm used in this system is inspired by the GMM-based speaker verification technique that was introduced in [Reynolds, 1995] and [Reynolds *et al.*, 2000]. This algorithm can be divided into two phases - the initialization phase and the iteration phase. The iteration phase tries to assign each audio segment to the speaker cluster that best represents the speaker. This is repeated until such a point where the cluster assignment is found to have stabilized between successive iterations. This process of iterative re-clustering has the effect of increasing the speaker homogeneity of each cluster.

- **Initialization phase**

Step 1: Beamforming & MFCC feature extraction

Beamforming is first performed using all the available source audio so as to generate an enhanced recording as described in [Anguera *et al.*, 2005a]. A voice activity detector (VAD) is then applied to retain only the high energy frames. MFCC acoustic features are then generated from the enhanced audio signal. These MFCC acoustic features will be used in Steps 2 to 5.

*It is to be noted that the modules in this section were developed by Dr. Sun Hanwu & Dr. Nwe Tin Lay of the Institute for Infocomm Research (I^{2}R), Agency For Science, Technology And Research (A*STAR), Singapore. Dr. Sun was responsible for the cluster purification module, while Dr. Nwe was responsible for the Non-speech and silence removal modules.



Step 1: All segments are used to train Θ_{Root}

Step 2: For each cluster $q = 1..Q$, Q = number of clusters, GMM $\Theta_{0,q}$ is adapted from Θ_{Root}

Step 3: All segments are scored against $\Theta_{i,1} \dots \Theta_{i,Q}$, $i = 0$ for initial iteration. Segments are then re-assigned to the cluster in which it scored highest.

Step 4: For $q = 1..Q$, re-assigned clusters are used to adapt $\Theta_{i+1,q}$ from $\Theta_{i,q}$.

Step 5: $i = i + 1$. Repeat from Step 3. Stop when assignment stabilizes

Figure 4.11: An illustration of the cluster purification process.

Step 2: Initial model training

All the MFCC vectors resulting from acoustic feature extraction are then used to train a root Gaussian Mixture Model (GMM), Θ_{Root}. The Θ_{Root} has 40 Gaussian components with full covariance matrices and was trained using the Expectation Maximization (EM) algorithm as described in [Reynolds *et al.*, 2000].

Segments resulting from bootstrap clustering are then pooled together according to their cluster assignments. Individual GMMs are adapted from Θ_{Root} for every cluster. Adaptation is performed on the weights, means and variances using the Maximum a Posteriori (MAP) approach [Reynolds *et al.*, 2000]. Thus if there are Q speaker clusters resulting from bootstrap clustering, there will be Q GMMs. We denote these Q GMMs as $\Theta_{i,q}$, where i indicates the iteration number and $q = 1..Q$ indicates the q^{th} speaker cluster. For the initialization step, $i = 0$.

- **Iteration phase**

Maximum Likelihood assignment

Step 3: Let $\mathbf{O}_j$ denote the set of feature vectors extracted from the j^{th} segment. Let $cluster_i[j] \in \{1..Q\}$ denote the cluster assignment of $\mathbf{O}_j$ for iteration i.

Every segment $\mathbf{O}_j$ is scored against models $\Theta_{i,q}$. Each segment is then relabeled by

$$cluster_{(i+1)}[j] = \arg\max_{q=1..Q}\{p(\mathbf{O}_j|\Theta_{i,q})\} \qquad \text{(Eqn. 4.12)}$$

Model adaptation

Step 4: The GMMs $\Theta_{(i+1),q}$ are MAP adapted from $\Theta_{i,q}$ using the segments $\mathbf{O}_j$ corresponding to labels $cluster_{(i+1)}[j] = q$.

Step 5: $i = i + 1$; Repeat from Step 3, until the cluster assignments have stabilized and do not vary for successive iterations. The cluster assignment for segments were found to converge typically within 20 iterations.

4.3.3.2 Non-Speech and Silence Removal (NS&SR)

A typical meeting recording will contain periods of silence where nobody is speaking. The duration of these periods of silence constitute about 7% of the total durations for the RT-06s and RT-07 meeting recordings. It is thus necessary to identify these periods of silence and remove them from the diarization. The same would apply to non-speech events such as coughs, laughter or breathing noises. These noises are present intermittently within speech segments and should not be included in the speaker diarization transcript.

- **Non-speech removal**

The acoustic features used in this stage are the Log Frequency Power Coefficients (LFPC) [Nwe *et al.*, 2003]. A total of 10 coefficients are extracted from each 20 ms frame with a 10 ms overlap between frames. A model based approach was then used to evaluate every segment. Speech and non-speech were modeled by two separate GMMs, Θ_S and Θ_N. A classification decision can be made for the j^{th} segment, $\mathbf{O}_j$ as follows.

$$p(\Theta_S|\mathbf{O}_j) \geq p(\Theta_N|\mathbf{O}_j) \rightarrow \text{speech} \qquad \text{(Eqn. 4.13)}$$

$$p(\Theta_S|\mathbf{O}_j) < p(\Theta_N|\mathbf{O}_j) \rightarrow \text{non} - \text{speech} \qquad \text{(Eqn. 4.14)}$$

When expressed as a likelihood ratio, Eqn. 4.13 and Eqn. 4.14 become

$$\frac{p(\mathbf{O}_j|\Theta_S)}{p(\mathbf{O}_j|\Theta_N)} \geq th_1 \rightarrow \text{speech} \qquad \text{(Eqn. 4.15)}$$

$$\frac{p(\mathbf{O}_j|\Theta_S)}{p(\mathbf{O}_j|\Theta_N)} < th_1 \rightarrow \text{non} - \text{speech} \qquad \text{(Eqn. 4.16)}$$

where $th_1 = \frac{p(\Theta_N)}{p(\Theta_S)}$ is a threshold that is trained on an external corpus.

- **Silence removal**

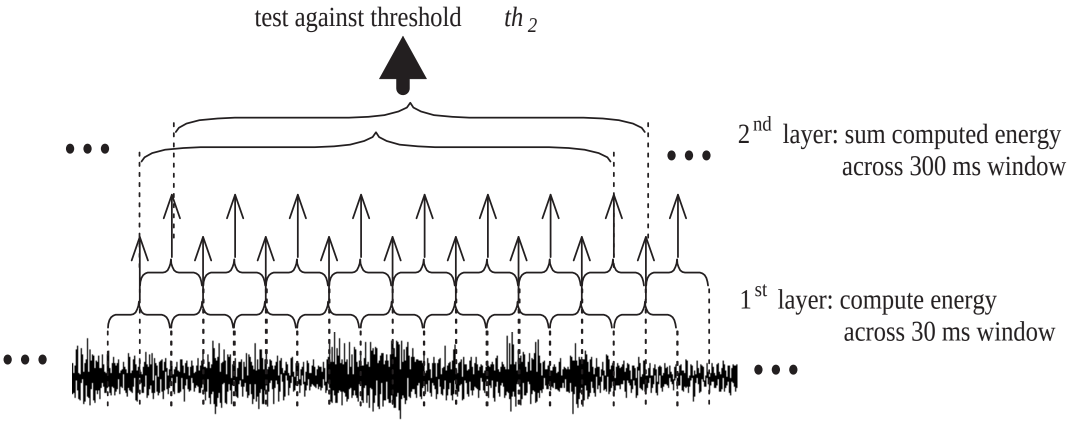

test against threshold th_2
2nd layer: sum computed energy across 300 ms window
1st layer: compute energy across 30 ms window

Figure 4.12: The "Double-Layer Windowing" method of removing silence.

Silence was removed using a "Double-Layer Windowing" method. In the first layer, audio is divided into frames of 20 ms with 10 ms overlapping. The energy for each frame is computed. In order to remove silences that are longer than the 300 ms tolerance specified for the evaluation, a second layer window is applied. This 300 ms long window shifts in 10 ms steps. The energy across all the frames in the window is summed. When this energy is found to cross a threshold, th_2, the region covered by the window will be deemed as silence and dropped.

4.4 System Performance

4.4.1 Results on RT-06s

Table 4.4 presents the system's performance for experiments conducted upon the RT-06s evaluation corpus. The system's performance was evaluated by computing the Diarization Error Rate (DER) against the official references released by NIST.

Table 4.4: DER for evaluations on the RT-06s corpus

RT-06s Task	† After				
	bootstrap clustering	cluster purification		non-speech & silence removal	
	DER (%)	DER (%)	% Δ	DER (%)	% Δ
CMU_20050912-0900	32.62	33.05	1.3	33.09	0
CMU_20050914-0900	27.85	27.05	-3	27.05	0
EDI_20050216-1051	25.75	26.09	1	25.62	-2
EDI_20050218-0900	22.60	22.94	2	22.93	0
NIST_20051024-0930	36.19	35.50	-2	35.50	0
NIST_20051102-1323	30.46	29.33	-4	29.32	0
VT_20050623-1400	59.49	40.05	-33	37.89	-5
VT_20051027-1400	44.55	43.59	-2	38.06	-13
Overall	34.19	31.87	-7	31.02	-3

†: These results were obtained using the official NIST RT-06s reference.

It was found that the DER improved after every successive processing stage. The overall DERs obtained were 34.19% after bootstrap clustering, 31.87% after cluster purification, and 31.02% after non-speech & silence removal (NS&SR). This overall DER of 31.02% was found to be better than other state-of-art systems reporting using the official NIST reference. As was listed in Table 4.2, the closest performing system [Janin *et al.*, 2006] achieved a DER of 35.80%.

Further observation of DERs for individual tasks indicates that cluster purification generally had a beneficial effect. DER improvements were observed for all tasks except those of EDI (*EDI_20050216-1051* & *EDI_20050218-0900*) and *CMU_20050912-0900* .

These tasks had absolute DER deteriorations of less than $\frac{1}{2}$%. In the tasks that showed improvements, most of these DER gains were of less than 1 absolute %. The biggest gains were obtained for *VT_20050623-1400* . Cluster purification gave that task an absolute DER boost of over 19 % points. The biggest DER gains for cluster purification can be seen for the *VT_20050623-1400* task. We concluded that the initial clustering using DOA for this task was unreliable as compared to the other tasks. Hence cluster purification using acoustic features was able to produce better clustering.

Marginal DER improvements were also obtained for the stage of Non-Speech & Silence removal (NS&SR). The overall DER dropped by 0.85% as a result of this stage.

4.4.2 Results on RT-07

This section examines the performance of the diarization system on the RT-07 evaluation corpus.

The DER for every task was found to improve after each processing step. The final DER of 15.32% was found to be competitive versus the other systems submitted for the RT-07 evaluation.

Table 4.5: DER for evaluations on the RT-07 corpus

		After			
	bootstrap clustering	cluster purification		non-speech & silence removal	
RT-07 Task	DER (%)	DER (%)	% Δ	DER (%)	% Δ
CMU_20061115-1030	22.75	22.84	+0.4	19.48	-15
CMU_20061115-1530	17.81	17.51	-2	12.46	-29
EDI_20061113-1500	24.29	22.91	-6	20.69	-10
EDI_20061114-1500	30.59	28.45	-7	15.00	-47
NIST_20051104-1515	23.34	22.45	-4	12.66	-44
NIST_20060216-1347	22.13	18.35	-17	13.36	-27
VT_20050408-1500	46.38	19.77	-57	11.32	-43
VT_20050425-1000	27.36	25.41	-7	18.45	-27
Overall	27.02	22.13	-18	15.32	-31

It was observed that after bootstrap clustering, the *VT_20050408-1500* task had the worst DER of 46.38%. A clue to its poor performance can be seen in the Speaker Error time (SE) component of the DER. It made up almost two-thirds of the diarization errors. It was observed that the TDOA estimation for this task was highly inaccurate. This resulted in many speech segments being incorrectly attributed to the wrong speaker. It is hypothesized that the cause of the TDOA inaccuracy is the high reverberation of the VT meeting room. The effect of reverberations on accuracy and diarization performance will be discussed in greater detail in Section 4.5.1.2. The DER for *VT_20050408-1500* improves considerably to 19.77% after performing cluster purification. This suggests that the cluster purification method is capable of redeeming speaker assignment errors introduced by inaccurate TDOA estimation.

It is noteworthy that after cluster purification, the improvements for most other tasks were marginal. This is because the Speaker Error time for these tasks were already rather low after bootstrap clustering and hence cluster purification thus had little room to improve upon.

Finally, the NS&SR stage also helped to produced big improvements of 31% DER reduction in the DER scores.

4.4.3 Number of speaker detected on RT-06s & RT-07

In an ideal clustering scenario, every speaker should be represented by one and only one cluster. Experimental results showed that our proposed system was capable of correctly estimating the number of speakers present in most of the RT-06s and RT-07 tasks. However when there were a larger number of speakers present, the system was observed to identify fewer speakers than in actual case. Amongst the RT-06s tasks, 2 tasks had their number of speakers present under-estimated. There are 9 and 8 speakers respectively in *NIST_20051024-0930* & *NIST_20051102-1323* . The system could identify only 5 and 6 speakers. Analysis showed that the missing speakers were those that spoke for the shortest durations. These missing speakers spoke for between 147 to 39 seconds.

Table 4.6: Actual number of speakers vs. number of speakers detected on the RT-06s & RT-07 corpora

	Actual # of speakers	# speakers detected
RT-06s tasks		
CMU_20050912-0900	4	4
CMU_20050914-0900	4	4
EDI_20050216-1051	4	4
EDI_20050218-0900	4	4
NIST_20051024-0930	9	5
NIST_20051102-1323	8	6
VT_20050623-1400	5	5
VT_20051027-1400	4	4
RT-07 tasks		
CMU_20061115-1030	4	4
CMU_20061115-1530	4	4
EDI_20061113-1500	5	4
EDI_20061114-1500	4	4
NIST_20051104-1515	4	4
NIST_20060216-1347	5	6
VT_20050408-1500	5	5
VT_20050425-1000	4	4

Inaccurate estimations in the number of speakers present were also observed for two tasks in the RT-07 corpus. The 8 tasks that made up RT-07 all had either 4 or 5 speakers present in the recordings. The *NIST_20060216-1347* task further confirmed the observations found for RT-06s, i.e. that speakers with short durations tend to be omitted from the speaker count. Only 5 speakers were detected when there should be 6. The speaker who spoke the least (59 seconds of speech) was found merged with another speaker who spoke for 128 seconds. In the case of *EDI_20061113-1500* , the system found 5 speakers when there should be 4. The speech of the longest speaker was found to be divided into two clusters, one containing 441 seconds of speech and other 86 seconds.

The under-estimation of the number of speakers for *NIST_20051024-0930* , *NIST_20051102-1323* & *NIST_20060216-1347* was observed to be due to a shortage of resolution in the histograms used to perform quantization in the bootstrap clustering process. An illustration of this problem can be seen in Fig. 4.8 for a pair of microphones from *NIST_20060216-1347* . 4 peaks representing 4 speakers were present when there were 6 speakers in actual fact. When a large number of speakers is present, the resolution of the histogram becomes inadequate. Adjacent speaker distributions tend to overlap. Those speakers who spoke the least had small bin counts and therefore did not register as a clustering centroid. They thus had a tendency to be absorbed into their neighbours.

It is noted that the tasks exhibiting under-estimation happened to be from the NIST conference room. This is an interesting observation that would warrant future study to see if the conference room perhaps had a bearing on the mis-estimation of the number of speakers.

4.5 Discussions about the Speaker Diarization Experiments

The experiments conducted earlier this section have shown that the Time Delay of Arrival (TDOA) is a viable source of information for use in the speaker diarization of meetings. The system previously described earlier in Section 4.3 was capable of producing results on the RT-06s & RT-07 corpora that are comparable to that reported by other state-of-the-art systems. For the RT-06s corpus, an overall Diarization Error Rate (DER) of 31.02% was obtained. This compares favourablely with the best result of 35.80% that was reported in [Janin *et al.*, 2006] (See Table 4.2 for other results). The DER of 15.32% obtained on the RT-07 corpus was also competitive versus that reported in the literature. The best reported result was 8.51% in [Wooters & Huijbregts, 2007] (See Table 4.3).

A further examination of the diarization performance reveals that while using the TDOA alone can yield fairly good diarization decisions, additional DER gains are obtained by the cluster purification step. This step was capable of producing a 7% and 18% relative performance gain respectively on the RT-06s & RT-07 (See Table 4.4 & 4.5). The subsequent step of non-speech & silence removal was also capable of producing additional improvements. Further relative DER improvements of 3% & 31% were respectively observed for the two corpora.

The system described was submitted for evaluation in the NIST RT-07 benchmarking exercise and obtained an overall second placing. This work also resulted in two conference papers by the author and the NTU/I^{2}R team. Said papers are listed in Appendix A.

There were numerous issues that affected the diarization performance of the system. These issues will be discussed in the following section.

4.5.1 Issues affecting system performance

4.5.1.1 Number of detected speakers

As was noted earlier in Section 4.4.3, the diarization system exhibited a tendency to misjudge the number of speakers present when there are a large number of speakers. There are a number of reasons why the number of speakers may be detected incorrectly during the bootstrap clustering step. In the *NIST_20051024-0930* , *NIST_20051102-1323* & *NIST_20060216-1347* tasks from the previous section, it was found that the number of speakers present tends to be under-estimated when there is insufficient resolution in the histogram used for quantization. When this happens, multiple adjacent speakers can become merged into a common centroid.

As our current system in the bootstrap approach only utilizes Time Delay of Arrival (TDOA) estimates, we may be able to improve its performance by also using acoustic features. The use of acoustic features has been demonstrated to be effective in [Anguera *et al.*, 2006b; Anguera *et al.*, 2007].

4.5.1.2 The effect of audio multi-paths

As was noted earlier in Section 4.4, the VT tasks *VT_20050623-1400* , *VT_20051027-1400* & *VT_20050408-1500* were noted to yield performance that is much poorer than average. Further analysis of the recordings for these tasks suggest that the poor performance could be due to the presence of reverberations or audio multi-paths.

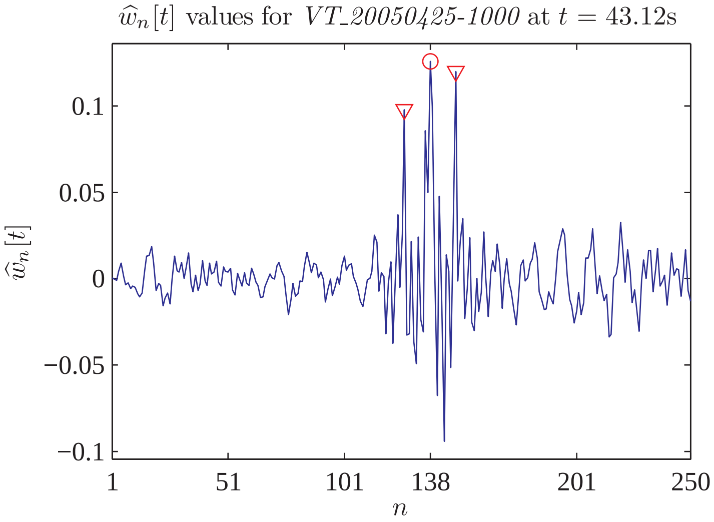


Figure 4.13: $\hat{\mathbf{w}}[t] = [\hat{w}_0[t] \cdots \hat{w}_n[t] \cdots \hat{w}_{L-1}[t]]^T$ values at an instance of reverberant speech. Three peaks can be observed at n = 127, 138 and 149. Triangle markers (▽) indicate peaks, the circle marker (○) marks the highest coefficient.

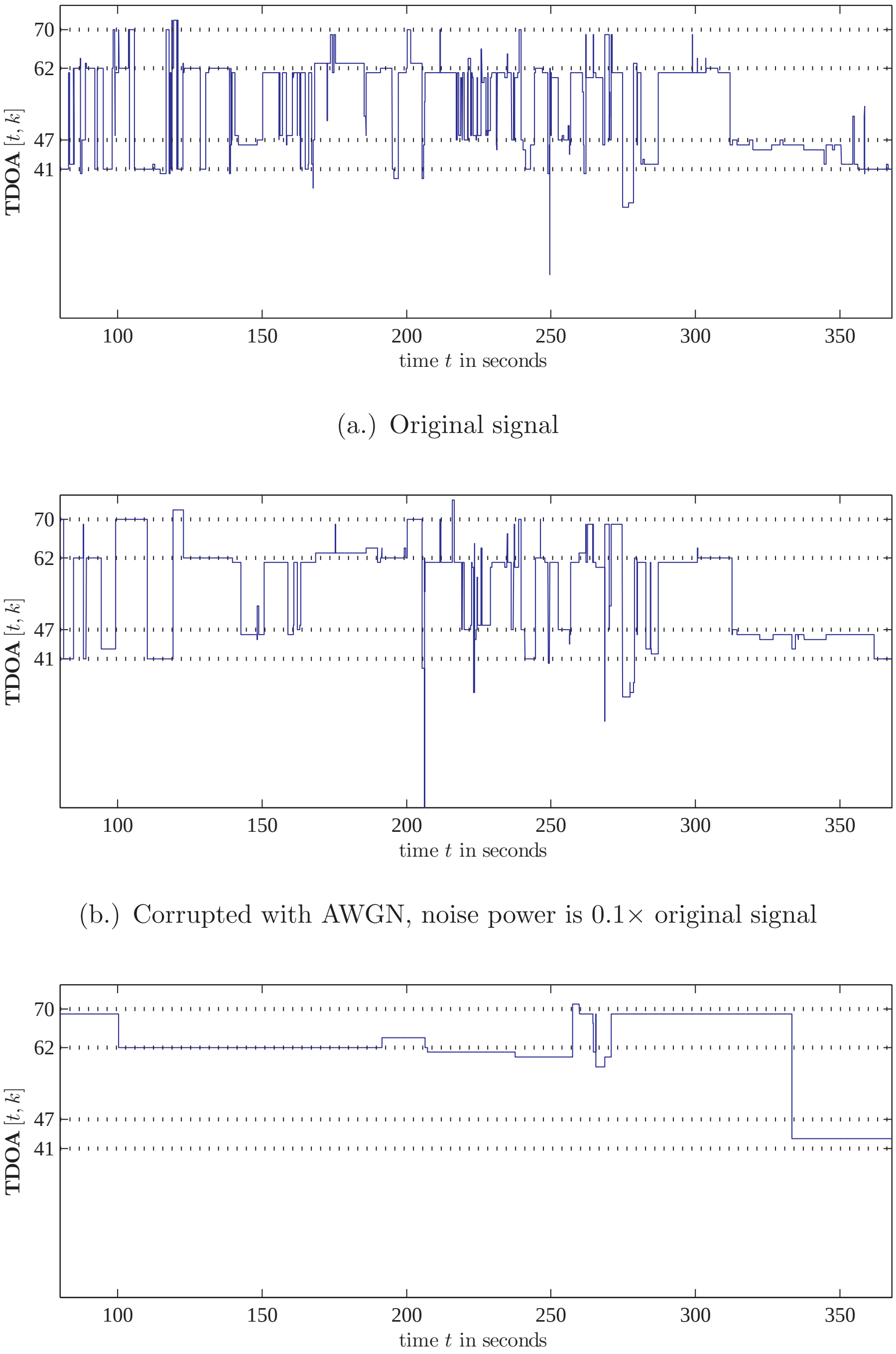
Fig. 4.13 shows a plot of the TDOA filter weights during a reverberant segment. There typically will be a small primary peak for the main path and numerous shorter peaks corresponding to the other multi-paths. Reverberations are the result of speech from the source being reflected off a solid surface, thus taking an indirect route to the microphones. Since the speech made by a speaker will travel to the microphones along many different paths, the distance traveled by the speech from the source to the microphones will vary for each path. The corresponding time delay of arrival (TDOA) between the two channels will thus be affected. This results in multiple TDOA values being found, as indicated by multiple peaks in the filter weights. Erroneous TDOA estimation results because the primary peak is weak. This thus allows secondary peaks to sometimes overwhelm the correct primary location.

4.5.1.3 The presence of noise

As is indicated in 4.3.1, when performing TDOA estimation using the NLMS filter, the source and reference channels (**s** [t] & **r** [t]) may be corrupted by additive noise. This noise will reduce the accuracy of the TDOA estimations. Fig. 4.14 shows a series of $\mathbf{TDOA}[t, k]$ plots for a segment of *CMU_20050912-0900* . White Gaussian noise is added in the **s** [t] and **r** [t] for plots (b.) and (c.). It can be seen that the presence of noise thus is detrimental to the estimation of TDOA.

The noise corrupting the microphone channels do not necessary have to be white and gaussian. In the paper [Brayda *et al.*, 2005], it was documented that frequency specific noise will also be of detriment to TDOA estimation. The NIST MarkIII [Rochet, 2005] microphone array setup was examined and it was found that frequency specific noise may be introduced by way of the microphone system collecting the recording. When the microphones are powered using alternating current running at 60Hz, noise may be introduced at the 60Hz frequency band. Noise pollution at other frequency bands can also be possible depending on the electronic circuitry used.

Frequency specific noise noise can be observed in the audio spectrogram as "lines" present at specific frequency bands. An analysis of the NIST RT audio recordings also suggest the presence of such frequency specific noise. Fig. 4.16 shows a spectrogram of the recording made by the 1^{st} distant microphone of *CMU_20061115-1030* . In it, a "line" can be observed at the 1.25 kHz, 3.3 kHz, 5.25 kHz and 6 kHz frequency bands. Similar artifacts can also be found in the recordings for the 2^{nd} and 3^{rd} distant microphones. This phenomena was observed to be not specific to the CMU meeting room as it was also present in the recordings for other meeting rooms, albeit at other frequency values.

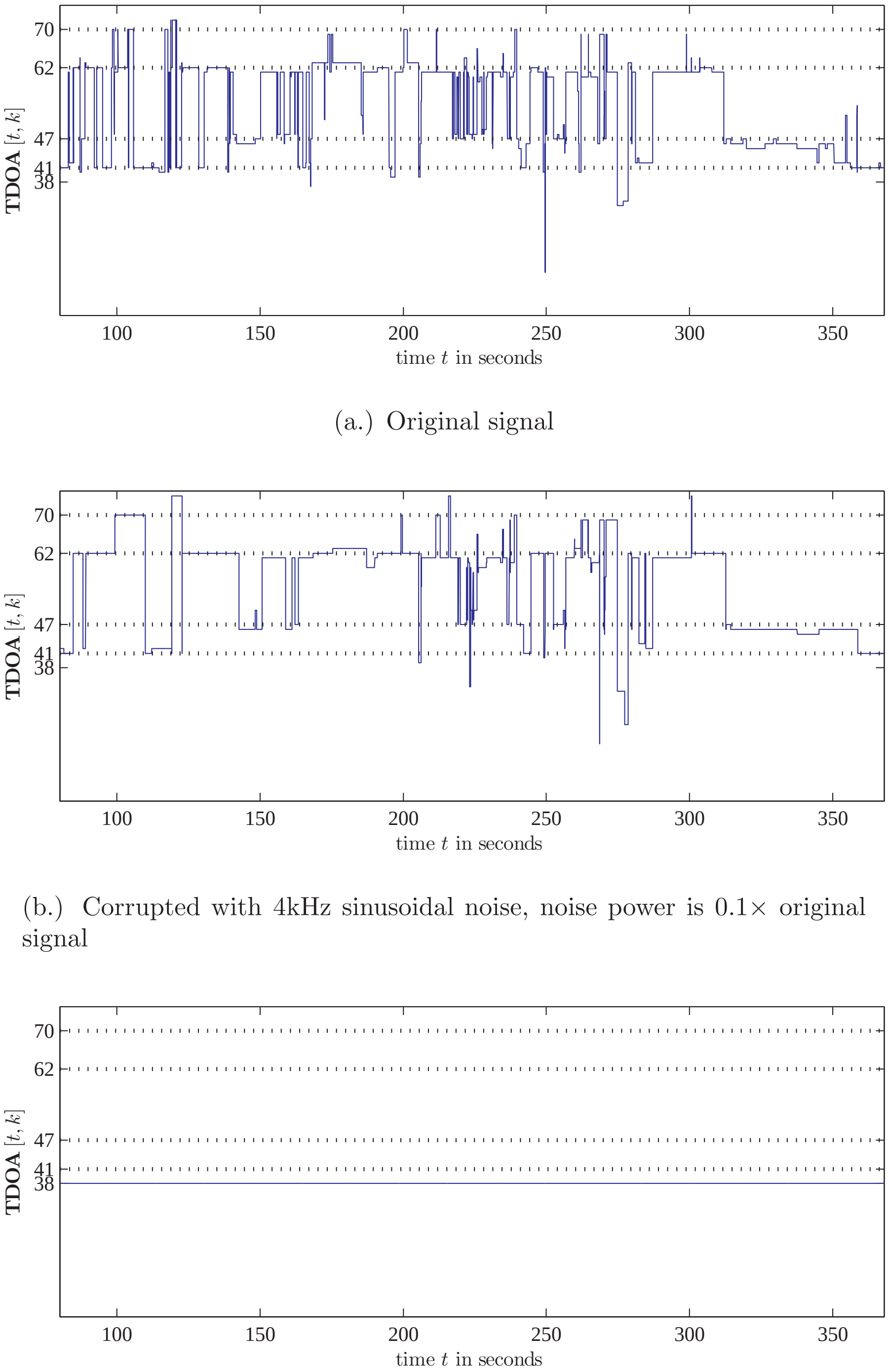


(a.) Original signal

(b.) Corrupted with AWGN, noise power is 0.1× original signal

(c.) Corrupted with AWGN, noise power is 2× original signal

Figure 4.14: For a segment of CMU_20050912-0900 , the **TDOA**$[t, k]$ *determined between the 1st and 2nd microphone pairs using the (a.) original recordings, (b.) corrupted with AWGN, noise power 0.1× that of original signal, (c.) corrupted with AWGN, noise power 2× that of original signal.*



(a.) Original signal

(b.) Corrupted with 4kHz sinusoidal noise, noise power is 0.1× original signal

(c.) Corrupted with 4kHz sinusoidal noise, noise power is 2× original signal

Figure 4.15*: For a segment of CMU_20050912-0900 , the* **TDOA**$[t, k]$ *determined between the 1*st *and 2*nd *microphone pairs using the (a.) original recordings, (b.) corrupted with 4kHz sinusoidal signal, noise power 0.1× that of original signal, (c.) corrupted with 4kHz sinusoidal signal, noise power 2× that of original signal.*

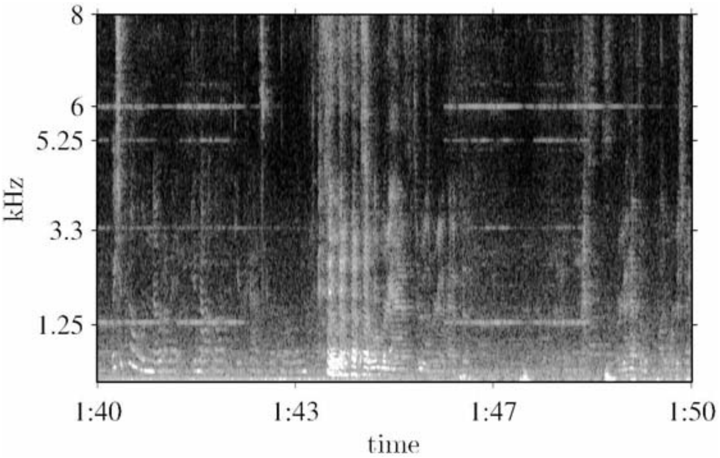
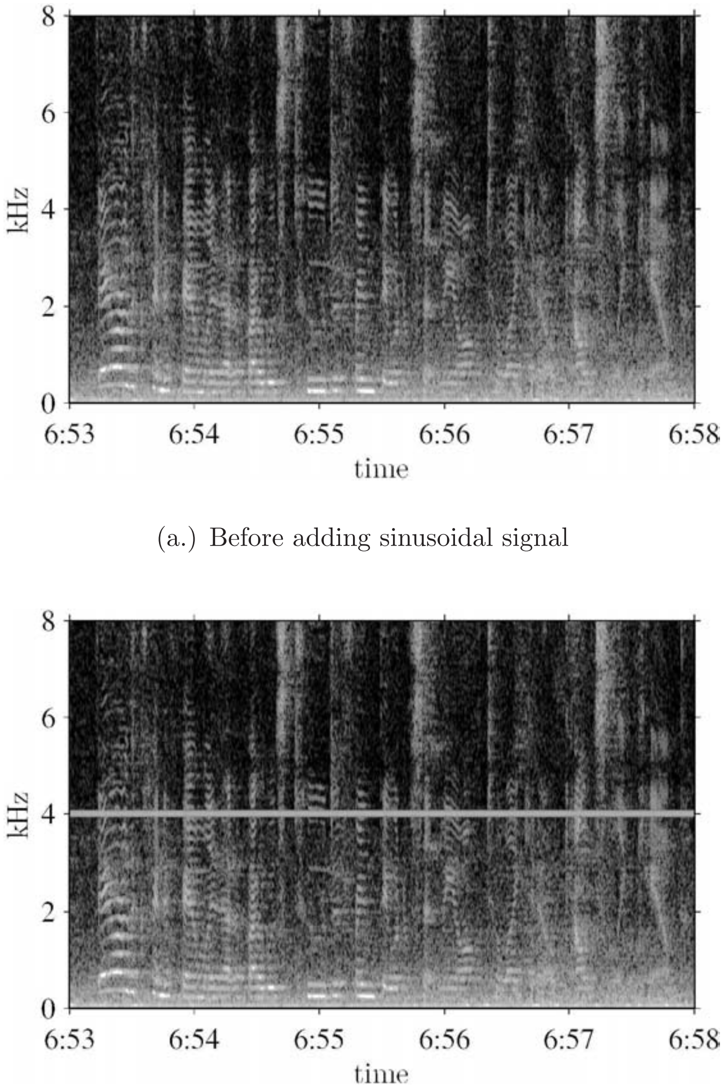


Figure 4.16: Spectrogram showing frequency specific noise inherent in CMU_20061115-1030 . Corruption can be found at 1.25 kHz, 3.3 kHz, 5.25 kHz and 6 kHz.

A simple experiment was done to confirm that frequency specific noise can indeed reduce diarization quality. Using the *CMU_20050912-0900* task from the RT-06s corpus, a sinusoidal signal was added at the 4 kHz band. Fig. 4.17 shows spectrograms of the audio recording before the addition of the sinusoid, and after the addition of the sinusoidal.

TDOA estimation when conducted upon the corrupted recording produces $\mathbf{TDOA}[t,k]$ time series plots as shown in Fig 4.15. It can be seen that there is a gradual deterioration in the quality of the estimation as the power of the corrupting signal is increased. When the corrupting signal power is increased to 2 times that of the original signal, the $\mathbf{TDOA}[t,k]$ value deteriorates to form a straight line at 38. This happens because when the power of corrupting signal is greater than that of the original signal and the corrupting signal becomes the dominant signal. The corrupting signal is sinusoidal and has the same phase delay for both channels. As there is no phase difference between the sinusoids added to both channels, no time delay is observed between the channels and the TDOA will reflect a value of $\frac{L}{2} = 38$. It is to be noted that a filter length L of 75



(a.) Before adding sinusoidal signal

(b.) After adding 4kHz sinusoidal signal

Figure 4.17: Spectrogram of a segment of CMU_20050912-0900 (a.) before adding sinusoidal signal (b.) after adding 4kHz sinusoidal signal.

was used in this experiment.

4.5.1.4 The effect of simultaneous overlapping speakers

The diarization system proposed in this thesis is not capable of handling segments containing simultaneous overlapping speakers. This however is a problem that will have to be addressed because it was found in [Shriberg *et al.*, 2001] that between 31% to 54% of sentence "spurts"* in meeting recordings contain some form of overlap between two or more speakers. A more recent study in [Leeuwen & Konecny, 2007] also found that the added step of addressing overlaps was capable of reducing the DER for their system by about 3.5% when evaluated upon the RT-07 corpora.

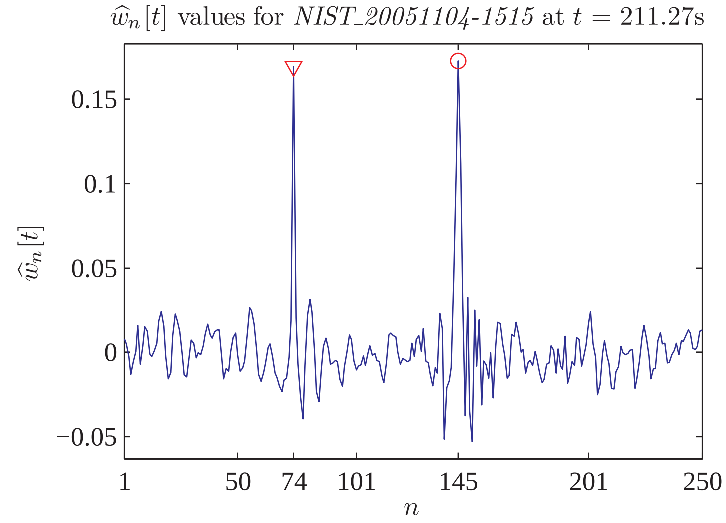


Figure 4.18: $\hat{\mathbf{w}}[t] = [\hat{w}_0[t] \cdots \hat{w}_n[t] \cdots \hat{w}_{L-1}[t]]^T$ *values at an instance where two speakers speak simultaneously. Two peaks can be observed at* $n = 74$ *and* 145*. Triangle markers (▽) indicate peaks, the circle marker (◯) marks the highest coefficient.*

An overlapping speech detection module can be used to determine if the voices of more than one speakers are present. Speech can then be properly attributed to the multiple individuals, thus reducing the SE component of the DER. Examples of such detection modules have been described in [Pfau *et al.*, 2001; Yamamoto *et al.*, 2006;

*A "spurt" is defined in [Shriberg *et al.*, 2001] to be a contiguous spoken sentence sub-set that is uninterrupted by pauses longer than 500 ms.

Fredouille & Senay, 2006] In the event that multiple speakers are present, multiple TDOA estimations can be made of the multiple peaks in $\hat{\mathbf{w}}[t]$. These TDOA estimates when used in the quantization steps described in Section 4.3.2 can lead to better speaker clustering decisions.

4.5.1.5 Inadequately long filter length

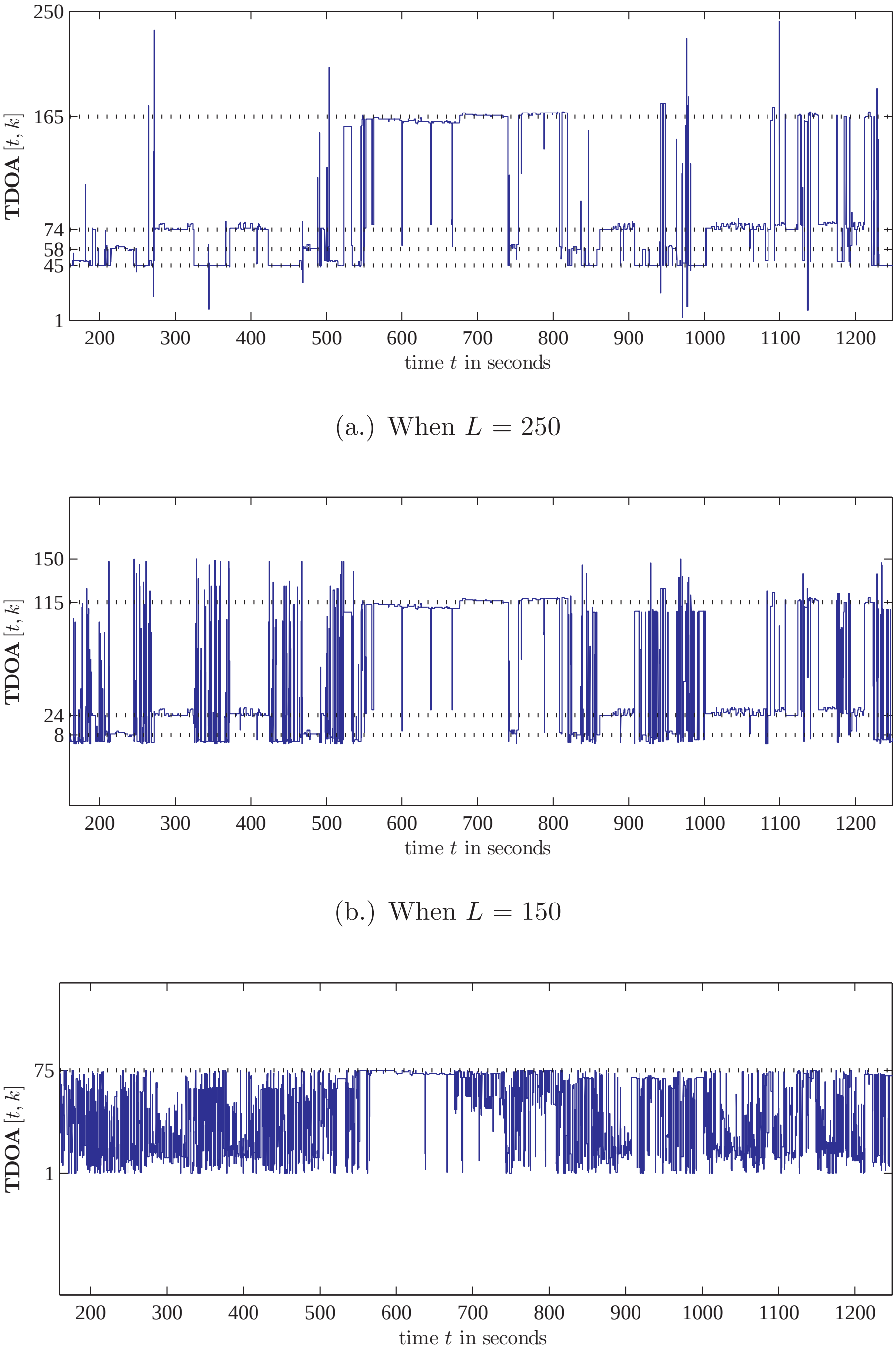
As was described in Section 4.3.1.1, a filter length $L = 250$ was used in the experiments on the RT-06s and RT-07 corpus. This value of L translates into a maximum microphone pair separation of 2.7 metres.

In cases where the microphone pair separation is greater than 2.7 metres, the TDOA estimation will fail. This is because the resultant delay of arrival between the two input channels will be greater than 7.8 ms. This delay will be out of the range detectable since the filter weights will not be able to register a corresponding primary peak. In such a situation, erroneous TDOA estimations will occur.

The following set of plots illustrates the problem described. Filter length of $L = 250$, 150 & 75 were used to estimate the TDOA for a segment of the *NIST_20051024-0930* task. Fig. 4.19 shows the $\mathbf{TDOA}(t, k)$ for the respective values of L. It can be seen that when the L used is inadequate, the TDOA estimations will be poor as values tend saturate at the limits of the dynamic range.

4.6 Summary of this chapter

This chapter proposed a speaker diarization system for meeting recordings that used a Normalized Least Means Squared (NLMS) filter to perform Time Delay of Arrival (TDOA) estimation. Bootstrap clustering then followed where the TDOA estimates were used as features to perform speaker segmentation and initial speaker clustering. This entailed performing two quantization steps (“within-pair” and “inter-pair”) using



(a.) When $L = 250$

(b.) When $L = 150$

(c.) When $L = 75$

Figure 4.19: For a segment of NIST_20051024-0930 , the $\mathbf{TDOA}[t, k]$ *determined between* 1^{st} *and* 3^{rd} *microphone pairs for (a.)* $L = 250$*, (b.)* $L = 150$ *and (c.)* $L = 75$*.*

histograms of the TDOA estimates and assigning audio frames to a number of initial clusters. These initial clusters were then purified using an iterative GMM-based approach, before Non-speech & Silence Removal (NS&SR) is performed.

The experiments conducted have shown the effectiveness of the TDOA as a source of information for speaker diarization. Results were obtained on the RT-06s & RT-07 corpora and the respective Diarization Error Rates (DERs) of 31.02% and 15.32% compares favourably against that reported by other state-of-the-art systems. Issues affecting the diarization performance of the system were also discussed and these issues will serve as room for future development of the system.

The next chapter will conclude this thesis and suggest some avenues for future work.

Chapter 5

Conclusions

5.1 Summary of results

In this thesis, the process of speaker diarization was explored. Investigation for the broadcast news domain concentrated on the speaker segmentation step while an integrated speaker diarization system was developed for the meeting room domain.

Speaker segmentation experiments were first carried out in Chapter 3 on the broadcast news domain in order to repeat the results reported in [Ajmera *et al.*, 2004] and [Zhou & Hansen, 2005]. In Section 3.2.4, the *BIC* was used as the divergence measure of choice. An optimal operating point was found where $\lambda = 0.5$ and experiments on the Hub4-97 corpus using said operating point produced *Recall*, *Precision* and *FScore* values of 0.639, 0.766 and 0.697 respectively. These results were comparable with that in [Ajmera *et al.*, 2004] which reported an *FScore* of 0.67.

Section 3.2.5 then carried out segmentation experiments using the T^2 divergence measure. A comparison of the resultant *BIC* and the T^2 time-series plots was done in Fig. 3.7. It can be seen that the T^2 curve identifies more spurious turns and is thus somewhat less accurate at detecting true turn points. This would translate to a higher turn point sensitivity (i.e. higher *Recall*) and lower *Precision* on the Hub4-97 corpus. The *Recall*, *Precision* and *FScore* at the optimal operating point for the T^2-based system was 0.556, 0.563 and 0.560 respectively.

The higher turn point sensitivity of the T^2 divergence measure was then used in Section 3.2.5.2 to complement the higher precision of the *BIC* divergence measure. A two-stage approach was used where the faster T^2 would shortlist points for re-evaluation by the *BIC*. This approach was first suggested in [Zhou & Hansen, 2000] and the combined system should have the advantage of being faster. The overall system developed for this thesis yielded *Recall*, *Precision* and *FScore* results of 0.641, 0.604 and 0.622 on the Hub4-97 corpus. These results were poorer than the *FScore* of 0.803 claimed in [Zhou & Hansen, 2005] and one reason for the poorer performance could be that the windowing algorithm used in the second stage of the combined system differed from that used in [Zhou & Hansen, 2005]. The experiments however did show the run time for the combined T^2+*BIC* system to be 1.4 times faster than the same for the *BIC*-based system.

Speaker diarization experiments were also conducted upon the NIST RT-06s & RT-07 meeting room corpora in Chapter 4. A speaker diarization system for meeting rooms was proposed where a Normalized Least Means Squared (NLMS) filter is used to perform Time Delay of Arrival (TDOA) estimation across different microphone pairs. The viability of performing diarization using the TDOA was shown. Using only this information, overall Diarization Error Rates (*DER*) of 31.02% & 15.32% were respectively obtained for the RT-06s & RT-07.

Diarization using only the TDOA however suffered from shortcomings resulting from inaccuracies when computing TDOA. Section 4.5.1 discussed some issues complicating this task. These limitations usually were due to sub-optimal meeting recordings (e.g.: when the recordings are corrupted by noise, when there are reverberations in the recordings), or otherwise due to the speaker locations in the room (e.g.: when speakers move about in the room, when multiple speakers are in close proximity to each other). Cluster purification was introduced to help overcome those clustering inaccuracies. It was found that with this step, the overall *DER*s for RT-06s & RT-07 could be improved by 7% & 18% respectively relative to the *DER*s without purification.

The additional steps of Non-Speech & Silence Removal (NS&SR) were then performed after cluster purification. These steps were done in order to better handle the nature of meeting recordings. Meeting recordings are usually characterized by having durations in which no participants are actively speaking, as well as the presence of non-speech events such as coughs, breathing noises or lip-smacks. A model based non-speech removal technique was employed along with an energy based silence removal step. This was found to further reduce the DERs by 3% & 31%.

5.2 Contributions

The contributions of this thesis are summarized as follows.

A speaker segmentation experiment was performed as described in Section 3.2.4 using the BIC divergence measure. A $FScore$ of 0.697 was obtained upon the Hub4-97e corpus and this score is comparable to the $FScore$ of 0.67 reported in [Ajmera *et al.*, 2004]. This experiment confirmed our implementation of the work described in [Ajmera *et al.*, 2004] and also showed the effectiveness of the BIC as a divergence measure for speaker segmentation.

A subsequent speaker segmentation experiment was carried out in Section 3.2.5.2 using a two-staged T^2+BIC algorithm. The results obtained however did not match that reported in [Zhou & Hansen, 2005] but timed runs of the experiment did show that a faster computational time could be achieved for such a fused system.

A speaker diarization system was proposed in Chapter 4 and experimental results were reported on the NIST RT-06s & RT-07 meeting room corpora. The diarization system used a Time Delay of Arrival (TDOA) based front-end with subsequent audio feature vector based cluster purification and non-speech & silence removal (NS&SR) steps. A NLMS filter was used to perform TDOA estimation for the specific purpose of speaker diarization. Results showed that the NLMS filter could effectively estimate the

TDOA between audio channels and that the TDOA is a viable source of information for use in estimating the location of speakers. A Diarization Error Rate (DER) of 31.02% was obtained on the RT-06s corpus and this compares favourably with the best result of 35.80% that was reported in [Janin *et al.*, 2006]. A DER of 15.32% was subsequently obtained on the RT-07 corpus and this result was found to be competitive versus the best reported result of 8.51% from [Wooters & Huijbregts, 2007]. Two conference papers resulted from this diarization system and said papers are listed in Appendix A.

A further analysis of issues affecting speaker diarization using the TDOA was done in Section 4.5.1. It is suggested that since the TDOA method estimates the spatial location of speakers in a meeting room, movements of individuals within the meeting room would lead to poor $DERs$. The presence of reverberations were also proposed to be detrimental to TDOA estimation, as are the presence of Gaussian and frequency specific noises in the meeting recordings.

5.3 Suggestions for future work

5.3.1 Adaptive peak detection threshold for BIC-based segmentation

The importance of the λ value to the segmentation performance of BIC was shown in Fig. 3.6. As the value of λ was varied, the segmentation performance would change abruptly. The performance "sweet spot" where the $FScore$ was above 0.6 was decidedly narrow (between $\lambda = 0.36$ to 0.6). The corpus specificity of the λ has also been reported in numerous papers [Tritschler & Gopinath, 1999; Lopez & Ellis, 2000; Vandecatseye & Martens, 2003; Ajmera *et al.*, 2004].

One way of overcoming this reliance on the λ might be to use an adaptive peak detection threshold th_{peak}. The current th_{peak} used is 0 and was held constant so that investigations into the performance of λ could be carried out. This form of an adaptive threshold had previously been explored in [Lu & Zhang, 2002b] for segmentation using the Divergence Shape Distance (DSD). In that paper, the threshold was set to be a running average of the DSD value. The same could also be done for the *BIC* or the T^2 divergence measures.

5.3.2 Using a first divergence measure to threshold another during segmentation

It has been concluded earlier in Section 3.2.5.1 that the T^2 tends to yield more spurious peaks while the *BIC* is more selective. One way of harnessing the relative strengths of these two divergence measures would be to use *BIC* to threshold the T^2 divergence curve. The relative scale of both divergence curves can be tweaked such that the T^2 curve will surpass that of the *BIC* when a true turn is encountered. The *KL2* has also been reported in [Siegler *et al.*, 1997; Kemp *et al.*, 2000] to be an algorithm that is more selective than the Mahalanobis or Bhattacharyya distances. This thus suggest possibilities of combining the T^2 or *BIC* with the *KL2* in order to make use of the relative strength of each method.

5.3.3 Improving the Non-Speech & Silence Removal (NS&SR) module for speaker diarization

As can be seen in Table 4.3, the meeting diarization system developed for this thesis performs worse than other reported systems when the $SAD\ DER$ is compared. The $SAD\ DER$ measures the Speech Activity Detection (SAD) performance of the system. This thus highlights a shortcoming of the system developed i.e. much non-speech & silence durations are not recognized by the NS&SR module. The current NS&SR module is only capable of reducing the $SAD\ DER$ from 14.45% to 8.65%.

A better silence detection module can thus be developed using other methods. One possibility is to perform silence detection using the signal entropy as opposed to energy. It was reported in [Waheed *et al.*, 2002] that entropy works better than energy for recordings made in a noisy environment because entropy tends to be more robust towards amplitude fluctuations.

Further gains may also be obtained by moving the NS&SR module to be ahead of the cluster purification step. The merit of doing so would be that cluster purification will be performed on segments that are free from silence and non-speech. The resultant speaker assignments may be more accurate, and this could be a way of further reducing the Speaker Error (SE) time component of the DER.

5.3.4 Better handling of multiple speaker instances in speaker diarization

It was described earlier in Section 4.5.1.4 that the TDOA estimation was difficult because instances of multiple concurrent speakers registered filter coefficients with multiple peaks. The filter coefficients at these instances appeared similar to the same for recordings made in an reverberant environment.

It would thus be beneficial to employ some form of multiple speaker detection in the system. Some of these techniques have been reported in works such as [Pfau *et al.*, 2001; Yamamoto *et al.*, 2006; Fredouille & Senay, 2006]. Instances with multiple concurrent speakers can be indicated as such and this would reduce the amount of Missed Speaker (MS) time in the DER.

5.3.5 Normalized Least Means Squared (NLMS) filter adaptation step size

An interesting avenue for future research would be to study the effect of the adaptation step size on the TDOA estimates produced by the NLMS filter. It is conceivable that a larger step size will serve to provide faster filter convergence and thus result in more accurate speaker segmentations, because speaker transitions are detected quicker.

A possible disadvantage of a larger step size however would be that it introduces greater fluctuations in the TDOA estimates. This could have ripple effects on the subsequent bootstrap clustering steps because both quantization steps (within-pair and inter-pair quantization) yield their best results when the TDOA estimates are stable.

Appendix A

List of Publications

- Koh, E. C. W. and Sun, H. and Nwe, T. L. and Nguyen, T. H. and Ma, B. and Chng, E. S. and Li, H. and Rahardja, S. ,“Speaker Diarization Using Direction of Arrival Estimate and Acoustic Feature Information: The I2R-NTU Submission for the NIST RT 2007 Evaluation”. In: *Rich Transcription 2007 Meeting Recognition Evaluation Workshop*, Baltimore, MD, USA: Springer LNCS 4625, 2007.

- Koh, E. C. W. and Sun, H. and Nwe, T. L. and Nguyen, T. H. and Ma, B. and Chng, E. S. and Li, H. and Rahardja, S. ,“Using Direction of Arrival Estimate and Acoustic Feature Information in Speaker Diarization”. In: *Interspeech’2007 - ICSLP - 10th International Conference on Spoken Language Processing*, Antwerp, Belgium, 2007.

References

Ajmera, J., & Wooters, C. 2003. A Robust Speaker Clustering Algorithm. *In: ASRU 2003 - 8th IEEE Automatic Speech Recognition and Understanding Workshop.*

Ajmera, J., McCowan, I., & Bourlard, H. 2004. Robust speaker change detection. *IEEE Signal Processing Letters*, **11**(8).

Akita, Yuya, & Kawahara, Tatsuya. 2003. Unsupervised Speaker Indexing Using Anchor Models and Automatic Transcription of Discussions. *In: Interspeech'2003 - 8th European Conference on Speech Communication and Technology.*

Anguera, X., Wooters, C., & Hernando, J. 2005a. Speaker diarization for multi-party meetings using acoustic fusion. *In: ASRU 2005 - 9th IEEE Automatic Speech Recognition and Understanding Workshop.*

Anguera, X., Wooters, C., Pardo, J., & Hernando, J. 2007. Automatic Weighting for the Combination of TDOA and Acoustic Features in Speaker Diarization for Meetings. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2007.*

Anguera, Xavier. 2005. *XBIC: Real-Time Cross Probabilities measure for speaker segmentation.* Tech. rept. ICSI.

Anguera, Xavier. 2006a (12 Feburary 2007). *ICSI-SRI forced alignments for the RT evaluations.*

Anguera, Xavier. 2006b. *Robust Speaker Diarization for Meetings.* Ph.D. thesis, Universitat Politecnica de Catalunya.

Anguera, Xavier, Wooters, Chuck, Peskin, Barbara, & Aguilo, Mateu. 2005b. Robust Speaker Segmentation for Meetings: The ICSI-SRI Spring 2005 Diarization System. *In: Rich Transcription 2005 Spring Meeting Recognition Evaluation Workshop*. Edinburgh, UK: Springer LNCS 3869.

Anguera, Xavier, Wooters, Chuck, & Hernando, Javier. 2006a. Friends and Enemies: A Novel Initialization for Speaker Diarization. *In: Interspeech'2006 - ICSLP - 9th International Conference on Spoken Language Processing*.

Anguera, Xavier, Wooters, Chuck, & Pardo, Jos M. 2006b. Robust Speaker Diarization for Meetings: ICSI RT06S Meetings Evaluation System. *In: Rich Transcription 2006 Spring Meeting Recognition Evaluation Workshop*. Bethesda, MD, USA: Springer LNCS 4299.

Barras, Claude, Zhu, Xuan, Meignier, Sylvain, & Gauvain, Jean-Luc. 2004. Improving Speaker Diarization. *In: Rich Transcription 2004 Fall Workshop*.

Barras, Claude, Zhu, Xuan, Meignier, Sylvain, & Gauvain, Jean-Luc. 2006. Multistage Speaker Diarization of Broadcast News. *IEEE Transactions on Audio, Speech, and Language Processing*, **14**(5).

Betser, Michael, Bimbot, Frdric, Ben, Mathieu, & Gravier, Guillaume. 2004. Speaker Diarization using bottom-up clustering based on a Parameter-derived Distance between adapted GMMs. *In: ICSLP 2004 - 8th International Conference on Spoken Language Processing*.

Bimbot, Frdric, & Mathan, Luc. 1993. Text-free speaker recognition using an arithmetic-harmonic sphericity measure. *In: Eurospeech'93 - 3rd European Conference on Speech Communication and Technology*.

Black, A., & Schultz, T. 2006. Speaker Clustering for Multilingual Synthesis. *In: ISCA Tutorial and Research Workshop on Multilingual Speech and Language Processing*.

Bonastre, J.-F., Delacourt, P., Fredouille, C., Merlin, T., & Wellekens, C. 2000. A speaker tracking system based on speaker turn detection for NIST evaluation. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2000*.

Brayda, Luca, Bertotti, Claudio, Cristoforetti, Luca, Omologo, Maurizio, & Svaizer, Piergiorgio. 2005. Modifications on NIST MarkIII Array to Improve Coherence Properties Among Input Signals. *In: AES 2005, 118th Audio Engineering Society Convention.*

Burges, C.J.C. 1998. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, **2**(2).

Cettolo, Mauro. 2000. Segmentation, Classification and Clustering of an Italian Broadcast News Corpus. *In: 6th RIAO 2000 - Content-Based Multimedia Information Access.*

Chen, Jingdong, Benesty, Jacob, & Huang, Yiteng. 2006. Time delay estimation in room acoustic environments: an overview. *EURASIP Journal on Applied Signal Processing*, **2006**(1).

Chen, Scott Shaobing, & Gopalakrishnan, P.S. 1998a. Clustering via the Bayesian information criterion with applications in speech recognition. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 1998.*

Chen, Scott Shaobing, & Gopalakrishnan, P.S. 1998b. Speaker, Environment And Channel Change Detection And Clustering Via The Bayesian Information Criterion. *In: DARPA Speech Recognition Workshop 1998.*

Cheng, E., Lukasiak, J., Burnett, I.S., & Stirling, D. 2005. Using spatial cues for meeting speech segmentation. *In: IEEE ICME'05 - International Conference on Multimedia and Expo 2005.*

Cohen, A., & Lapidus, V. 1995. Unsupervised text independent speaker classification. *In: 18th Convention of Electrical and Electronics Engineers in Israel.*

Cook, G. D., & Robinson, A. J. 1998. The 1997 Abbot system for the transcription of broadcast news. *In: DARPA Speech Recognition Workshop 1998.*

Couvreur, L., & Boite, J. 1999. Speaker Tracking in Broadcast Audio Material in the Framework of the THISL Project. *In: ESCA Tutorial and Research Workshop Accessing Information in Spoken Audio.*

Davis, S. B., & Mermelstein, P. 1980. Comparison of Parametric Representation for Monosyllabic Word Recognition in Continuously Spoken Sentences. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **28**(4).

Delacourt, P., & Wellekens, C. J. 2000. DISTBIC: A speaker-based segmentation for audio data indexing. *Speech Communication*, **32**(1-2).

Deng, Jing, Zheng, Thomas Fang, & Wu, Wenhu. 2006. UBM Based Speaker Segmentation and Clustering for 2-Speaker Detection. *In: ISCSLP 2006 - 5th International Symposium for Chinese Spoken Language Processing*.

Doco-Fernndez, L., & Garca-Mateo, C. 2005. Speaker Segmentation, Detection and Tracking in Multi-Speaker Long Audio Recordings. *In: Third COST275 Workshop - Biometrics on the Internet*.

Duda, Richard O., Hart, Peter E., & Stork, David G. 2000. *Pattern Classification, Second Edition*. Second edition edn. John Wiley & Sons.

Ellis, D., & Liu, J. 2004. Speaker turn segmentation based on between-channel differences. *In: ICASSP-NIST Meeting Recognition Workshop 2004*.

Fiscus, Jonathan G., Radde, Nicolas, Garofolo, John S., Le, Audrey, Ajot, Jerome, & Laprun, Christophe. 2005. The Rich Transcription 2005 Spring Meeting Recognition Evaluation. *In: Rich Transcription 2005 Spring Meeting Recognition Evaluation Workshop*. Edinburgh, UK: Springer LNCS 3869.

Fiscus, Jonathan G., Ajot, Jerome, Michel, Martial, & Garofolo, John S. 2006. The Rich Transcription 2006 Spring Meeting Recognition Evaluation. *In: Rich Transcription 2006 Spring Meeting Recognition Evaluation Workshop*. Bethesda, MD, USA: Springer LNCS 4299.

Fiscus, Jonathan G., Ajot, Jerome, & Garofolo, John S. 2007. The Rich Transcription 2007 Meeting Recognition Evaluation. *In: Rich Transcription 2007 Meeting Recognition Evaluation Workshop*. Baltimore, MD, USA: Springer LNCS 4625.

Fredouille, C., Moraru, D., Meignier, S., Besacier, L., & Bonastre, J.-F. 2004. The NIST 2004 spring rich transcription evaluation : two-axis merging strategy in the context of multiple distance microphone based meeting speaker segmentation. *In: Rich Transcription 2004 Spring Workshop.*

Fredouille, Corinne, & Evans, Nicholas. 2007. The LIA RT'07 speaker diarization system. *In: Rich Transcription 2007 Meeting Recognition Evaluation Workshop.* Baltimore, MD, USA: Springer LNCS 4625.

Fredouille, Corinne, & Senay, Grgory. 2006. Technical Improvements of the E-HMM Based Speaker Diarization System for Meeting Records. *In: Rich Transcription 2006 Spring Meeting Recognition Evaluation Workshop.* Bethesda, MD, USA: Springer LNCS 4299.

Gauvain, L., Lamel, L., & Adda, G. 1998. Partitioning and transcription of broadcast news data. *In: ICSLP 1998 - 5th International Conference on Spoken Language Processing.*

Gish, H., Siu, M. H., & Rohlicek, R. 1991. Segregation of speakers for speech recognition and speaker identification. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 1991.*

Graff, D., Fiscus, J., & Garofolo, J. 2002. *1997 HUB4 English Evaluation Speech and Transcripts.*

Grebenskaya, Olga, Kinnunen, Tomi, & Frnti, Pasi. 2005. Speaker Clustering in Speech Recognition. *In: 2005 Finnish Signal Processing Symposium.*

Hain, T., & Woodland, P.C. 1998. Segmentation and Classification of Broadcast News Audio. *In: ICSLP 1998 - 5th International Conference on Spoken Language Processing.*

Hain, Thomas, Johnson, Sue, Tuerk, Andreas, Woodland, Philip, & Young, Steve. 1998. Segment Generation and Clustering in the HTK Broadcast News Transcription System. *In: DARPA Broadcast News Transcription and Understanding Workshop.*

Hain, Thomas, Burget, Lukas, Dines, John, McCowan, Iain, Garau, Giulia, Karafit, Martin, Lincoln, Mike, Moore, Darren, Wan, Vincent, Ordelman, Roeland, & Renals, Steve. 2005. The Development of the AMI System for the Transcription of Speech in Meetings. *In: Rich Transcription 2005 Spring Meeting Recognition Evaluation Workshop*. Edinburgh, UK: Springer LNCS 3869.

Hain, Thomas, Burget, Lukas, Dines, John, Garau, Giulia, Karafit, Martin, Lincoln, Mike, Vepa, Jithendra, & Wan, Vincent. 2006. The AMI Meeting Transcription System: Progress and Performance. *In: Rich Transcription 2006 Spring Meeting Recognition Evaluation Workshop*. Bethesda, MD, USA: Springer LNCS 4299.

Hamming, Richard W. 1989. *Digital Filters (Third Edition)*. Third edition edn. Dover Publications.

Hansen, J.H.L., J.R. Deller, Jr., & Seadle, M.S. 2001. Transcript-Free Search of Audio Archives for the National Gallery of the Spoken Word. *In: JCDL 2001: Joint Conference on Digital Libraries 2001*.

Harris, Matthew, Aubert, Xavier, Haeb-Umbach, Reinhold, & Beyerlein, Peter. 1999. A Study of Broadcast News Audio Stream Segmentation and Segment Clustering. *In: Eurospeech'99 - 6th European Conference on Speech Communication and Technology*.

Haubold, Alexander, & Kender, John R. 2006. *Accommodating Sample Size Effect on Similarity Measures in Speaker Clustering*. Tech. rept. Department of Computer Science, Columbia University.

Haykin, Simon. 2001. *Adaptive Filter Theory, Fourth Edition*. Fourth edition edn. Prentice Hall.

Heck, Larry, & Sankar, Ananth. 1997. Acoustic Clustering and Adaptation for Robust Speech Recognition. *In: Eurospeech'97 - 5th European Conference on Speech Communication and Technology*.

Hermansky, Hynek. 1990. Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustical Society of America*, **87**(4).

Hotelling, Harold. 1931. The Generalization of Student's Ratio. *The Annals of Mathematical Statistics*, **2**(3).

Huang, R., & Hansen, J.H.L. 2004. Advances in unsupervised audio segmentation for the broadcast news and NGSW corpora. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2004.*

Huang, Rongqing, & Hansen, J. H. L. 2006. Advances in unsupervised audio classification and segmentation for the broadcast news and NGSW corpora. *IEEE Transactions on Audio, Speech, and Language Processing*, **14**(3).

Hung, Jeih-Weih, Wang, Hsin-Min, & Lee, Lin-Shan. 2000. Automatic Metric-Based Speech Segmentation for Broadcast News via Principal Component Analysis. *In: ICSLP 2000 - 6th International Conference on Spoken Language Processing.*

Istrate, Dan, Fredouille, Corinne, Meignier, Sylvain, Besacier, Laurent, & Bonastre, Jean-Franois. 2005. NIST RT'05S Evaluation: Pre-processing Techniques and Speaker Diarization on Multiple Microphone Meetings. *In: Rich Transcription 2005 Spring Meeting Recognition Evaluation Workshop.* Edinburgh, UK: Springer LNCS 3869.

Janin, A., Stolcke, A., Anguera, X., Boakye, K., Cetin, O., Frankel, J., & Zheng, J. 2006. The ICSI-SRI Spring 2006 Meeting Recognition System. *In: Rich Transcription 2006 Spring Meeting Recognition Evaluation Workshop.* Bethesda, MD, USA: Springer LNCS 4299.

Jin, Hubert, Kubala, Francis, & Schwartz, Rich. 1997. Automatic Speaker Clustering. *In: DARPA Speech Recognition Workshop 1997.*

Jin, Qin, & Schultz, Tanja. 2004. Speaker Segmentation and Clustering in Meetings. *In: ICSLP 2004 - 8th International Conference on Spoken Language Processing.*

Jin, Qin, Laskowski, Kornel, Schultz, Tanja, & Waibel, Alex. 2004. Speaker Segmentation and Clustering in Meetings. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2004.*

Johnson, S.E. 1999. Who Spoke When? Automatic Segmentation And Clustering For Determining Speaker Turns. *In: Eurospeech'99 - 6th European Conference on Speech Communication and Technology.*

Johnson, S.E., & Woodland, P.C. 1998. Speaker Clustering Using Direct Maximisation Of The MLLR-Adapted Likelihood. *In: ICSLP 1998 - 5th International Conference on Spoken Language Processing.*

Kaiser, J. F. 1990. On a simple algorithm to calculate the 'energy' of a signal. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 1990.*

Kemp, Thomas, Schmidt, Michael, Westphal, Martin, & Waibel, Alex. 2000. Strategies for Automatic Segmentation of Audio Data. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2000.*

Kim, Hyoung-Gook, Ertelt, D., & Sikora, T. 2005. Hybrid Speaker-Based Segmentation System Using Model-Level Clustering. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2005.*

Kimber, Donald G., Wilcox, Lynn D., Chen, Francine R., & Moran, Thomas P. 1995. Speaker segmentation for browsing recorded audio. *In: Conference on Human Factors in Computing Systems.*

Kinnunen, Tomi, Karpov, E., & Franti, P. 2006. Real-time speaker identification and verification. *IEEE Transactions on Audio, Speech, and Language Processing*, **14**(1).

Koh, E. C. W., Sun, H., Nwe, T. L., Nguyen, T. H., Ma, B., Chng, E. S., Li, H., & Rahardja, S. 2007a. Speaker Diarization Using Direction of Arrival Estimate and Acoustic Feature Information: The I2R-NTU Submission for the NIST RT 2007 Evaluation. *In: Rich Transcription 2007 Meeting Recognition Evaluation Workshop.* Baltimore, MD, USA: Springer LNCS 4625.

Koh, E. C. W., Sun, H., Nwe, T. L., Nguyen, T. H., Ma, B., Chng, E. S., Li, H., & Rahardja, S. 2007b. Using Direction of Arrival Estimate and Acoustic Feature Information in Speaker Diarization. *In: Interspeech'2007 - ICSLP - 10th International Conference on Spoken Language Processing.*

Kohonen, T. 1990. The self-organizing map. *Proceedings of the IEEE*, **78**(9).

Kubala, Francis, Davenport, Jason, Jin, Hubert, Liu, Daben, Leek, Tim, Matsoukas, Spyros, Miller, David, Nguyen, Long, Richardson, Fred, Schwartz, Richard, & Makhoul, John. 1998. The 1997 BBN Byblos System Applied To Broadcast News Transcription. *In: DARPA Speech Recognition Workshop 1998.*

Kwon, Soonil, & Narayanan, Shrikanth. 2002. Speaker change detection using a new weighted distance measure. *In: ICSLP 2002 - 7th International Conference on Spoken Language Processing.*

Lapidot, I., Guterman, H., & Cohen, A. 2002. Unsupervised speaker recognition based on competition between self-organizing maps. *IEEE Transactions on Neural Networks*, **13**(4).

Lapidot, Itshak. 2003. SOM as Likelihood Estimator for Speaker Clustering. *In: Interspeech'2003 - 8th European Conference on Speech Communication and Technology.*

Leeuwen, David A. van, & Huijbregts, Marijn. 2006. The AMI Speaker Diarization System for NIST RT06s Meeting Data. *In: Rich Transcription 2006 Spring Meeting Recognition Evaluation Workshop*. Bethesda, MD, USA: Springer LNCS 4299.

Leeuwen, David A. van, & Konecny, Matej. 2007. Progress in the AMIDA speaker diarization system for meeting data. *In: Rich Transcription 2007 Meeting Recognition Evaluation Workshop*. Baltimore, MD, USA: Springer LNCS 4625.

Linde, Y., Buzo, A., & Gray, R. 1980. An Algorithm for Vector Quantizer Design. *IEEE Transactions on Communications*, **28**(1).

Liu, Daben, & Kubala, Francis. 1999. Fast Speaker Change Detection for Broadcast News Transcription and Indexing. *In: Eurospeech'99 - 6th European Conference on Speech Communication and Technology.*

Lopez, Javier Ferreiros, & Ellis, Daniel P. W. 2000. Using Acoustic Condition Clustering To Improve Acoustic Change Detection On Broadcast News. *In: ICSLP 2000 - 6th International Conference on Spoken Language Processing.*

Lu, Lie, & Zhang, Hong-Jiang. 2002a. Real-time unsupervised speaker change detection. *In: 16th International Conference on Pattern Recognition.*

Lu, Lie, & Zhang, Hong-Jiang. 2002b. Speaker change detection and tracking in real-time news broadcasting analysis. *In: ACM International Conference on Multimedia 2002.*

Lu, Lie, Li, Stan Z., & Zhang, Hong-Jiang. 2001a. Content-Based Audio Segmentation Using Support Vector Machines. *In: ACM International Conference on Multimedia 2001.*

Lu, Lie, Jiang, Hao, & Zhang, Hong Jiang. 2001b. A Robust Audio Classification and Segmentation Method. *In: ACM International Conference on Multimedia 2001.*

Luque, Jordi, Anguera, Xavier, Temko, Andrey, & Hernando, Javier. 2007. Speaker Diarization for Conference Room: The UPC RT07s Evaluation System. *In: Rich Transcription 2007 Meeting Recognition Evaluation Workshop.* Baltimore, MD, USA: Springer LNCS 4625.

Meignier, S., Bonastre, J.-F., Fredouille, C., & Merlin, T. 2000. Evolutive HMM for multi-speaker tracking system. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2000.*

Meignier, Sylvain, Bonastre, Jean-Francois, & Igounet, Stephane. 2001. E-HMM approach for learning and adapting sound models for speaker indexing. *In: IEEE Odyssey Speaker Recognition Workshop 2001.*

Meignier, Sylvain, Bonastre, Jean-Franois, & Magrin-Chagnolleau, Ivan. 2002. Speaker Utterances Tying Among Speaker Segmented Audio Documents Using Hierarchical Classification: Towards Speaker Indexing of Audio Databases. *In: ICSLP 2002 - 7th International Conference on Spoken Language Processing.*

Meignier, Sylvain, Moraru, D., Fredouille, C., Besacier, L., & Bonastre, J.-F. 2004. Benefits of prior acoustic segmentation for automatic speaker segmentation. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2004.*

Meinedo, Hugo, & Neto, Joao. 2003a. Audio Segmentation, Classification and Clustering in a Broadcast News Task. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2003.*

Meinedo, Hugo, & Neto, Joo. 2003b. Automatic Speech Annotation and Transcription in a Broadcast News Task. *In: MSDR'2003 - ISCA Workshop on Multilingual Spoken Document Retrieval.*

Moh, Y., Nguyen, P., & Junqua, J.-C. 2003. Towards domain independent speaker clustering. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2003.*

Montaci, Claude, & Caraty, Marie-Jos. 1998. A Silence/Noise/Music/Speech Splitting Algorithm. *In: ICSLP 1998 - 5th International Conference on Spoken Language Processing.*

Moraru, D., Meignier, S., Besacier, L., Bonastre, J.-F., & Magrin-Chagnolleau, Y. 2003. The ELISA consortium approaches in speaker segmentation during the NIST 2002 speaker recognition evaluation. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2003.*

Mori, K., & Nakagawa, S. 2001. Speaker change detection and speaker clustering using VQ distortion for broadcast news speech recognition. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2001.*

Nishida, M., & Ariki, Y. 1999. Speaker indexing for news articles, debates and drama inbroadcasted TV programs. *In: IEEE International Conference on Multimedia Computing and Systems 1999.*

Nishida, M, & 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 2003.*

Nishida, M., & Kawahara, T. 2004. Speaker indexing and adaptation using speaker clustering based on statistical model selection. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2004.*

Nishida, M., & Kawahara, T. 2005. Speaker model selection based on the Bayesian information criterion applied to unsupervised speaker indexing. *IEEE Transactions on Speech and Audio Processing*, **13**(4).

NIST. 2004. *Fall 2004 Rich Transcription (RT-04F) Evaluation Plan.*

NIST. 2006a (29 November 2007). *NIST / SEMATECH e-Handbook of Statistical Methods.*

NIST. 2006b. *The NIST Year 2006 Speaker Recognition Evaluation Plan.*

NIST. 2006c. *Spring 2006 (RT-06S) Rich Transcription Meeting Recognition Evaluation Plan.*

NIST. 2007. *Spring 2007 (RT-07) Rich Transcription Meeting Recognition Evaluation Plan.*

Nwe, T. L., Foo, S. W., & Silva, L. C. D. 2003. Stress Classification Using Subband Based Features. *IEICE Transactions on Information and Systems*, **E86-D**(3).

Ore, B.M., Slyh, R.E., & Hansen, E.G. 2006. Speaker Segmentation and Clustering using Gender Information. *In: IEEE Odyssey Speaker and Language Recognition Workshop 2006.*

Pallett, David S., Fiscus, Jonathan G., Martin, Alvin, & Przybocki, Mark A. 1998. 1997 Broadcast News Benchmark Test Results: English and Non-English. *In: DARPA Speech Recognition Workshop 1998.*

Pardo, J.M., Anguera, X, & Wooters, C. 2006. Speaker Diarization for Multi-Microphone Meetings Using Only Between-Channel Differences. *In: 3rd Joint Workshop on Machine Learning and Multimodal Interaction (MLMI) 2006.* Bethesda, MD, USA: Lecture Notes in Computer Science 4299 Springer.

Pfau, Thilo, Ellis, Daniel, & Stolcke, Andreas. 2001. Multispeaker Speech Activity Detection For The ICSI Meeting Recorder. *In: ASRU 2001 - 7th IEEE Automatic Speech Recognition and Understanding Workshop.*

Pusateri, Ernest J., & Hazen, Timothy J. 2002. Rapid Speaker Adaptation Using Speaker Clustering. *In: ICSLP 2002 - 7th International Conference on Spoken Language Processing.*

Reed, F. A., Feintuch, P. L., & Bershad, N. J. 1981. Time delay estimation using the LMS adaptive filter - static behavior. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **29**(3).

Remes, U., Pylkkonen, J., & Kurimo, M. 2007. Segregation of Speakers for Speaker Adaptation in TV News Audio. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2007.*

Rentzeperis, Elias, Stergiou, Andreas, Boukis, Christos, Pnevmatikakis, Aristodemos, & Polymenakos, Lazaros C. 2006. The 2006 Athens Information Technology Speech Activity Detection and Speaker Diarization Systems. *In: Rich Transcription 2006 Spring Meeting Recognition Evaluation Workshop.* Bethesda, MD, USA: Springer LNCS 4299.

Reynolds, D.A., & Rose, R.C. 1995. Robust text-independent speaker identification using Gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, **3**(1).

Reynolds, D.A., Quatieri, T.F., & Dunn, R.B. 2000. Speaker Verification Using Adapted Gaussian Mixture Models. *Digital Signal Processing*, **10**(1).

Reynolds, Douglas A. 1995. Speaker identification and verification using Gaussian mixture speaker models. *Speech Communication*, **17**(1-2).

Rochet, Cedrick. 2005. *Technical Documentation of the Microphone Array Mark III.* Tech. rept. Information Access Division, National Institute of Standards and Technology, USA.

Rodrguez, Luis Javier, & Torres, M. Ins. 2004. A Speaker Clustering Algorithm for Fast Speaker Adaptation in Continuous Speech Recognition. *In: TSD 2004: Text, Speech and Dialogue.*

Rougui, J.E., Rziza, M., Aboutajdine, D., Gelgon, M., & Martinez, J. 2006. Fast Incremental Clustering of Gaussian Mixture Speaker Models for Scaling up Retrieval In On-Line Broadcast. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2006.*

Sankar, Ananth, Beaufays, Frangoise, & Digalakis, Vassilios. 1995. Training Data Clustering for Improved Speech Recognition. *In: Eurospeech'95 - 4th European Conference on Speech Communication and Technology.*

Scheffer, N., & Bonastre, J.-F. 2006. UBM-GMM Driven Discriminative Approach for Speaker Verification. *In: IEEE Odyssey Speaker and Language Recognition Workshop 2006.*

Schwarz, Gideon. 1978. Estimating the Dimension of a Model. *The Annals of Statistics*, **6**(2).

Shriberg, Elizabeth, Stolcke, Andreas, & Baron, Don. 2001. Observations on Overlap: Findings and Implications for Automatic Processing of Multi-Party Conversation. *In: Interspeech'2001 - 7th European Conference on Speech Communication and Technology.*

Siegler, M., Jain, U., Raj, B., & Stern, R. 1997. Automatic segmentation, classification and clustering of broadcast news audio. *In: DARPA Speech Recognition Workshop 1997.*

Siu, M. H., Yu, G., & Gish, H. 1992. An unsupervised, sequential learning algorithm for the segmentation of speech waveforms with multiple speakers. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 1992.*

Solomonoff, A., Mielke, A., Schmidt, M., & Gish, H. 1998. Clustering speakers by their voices. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 1998.*

Stadelmann, T., & Freisleben, B. 2006. Fast and Robust Speaker Clustering Using the Earth Mover's Distance and Mixmax Models. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2006.*

Stolcke, A., Anguera, X., Boakye, K., Cetin, O., Grezl, F., Janin, A., Mandal, A., Peskin, B., Wooters, C., & Zheng, J. 2005. Further Progress in Meeting Recognition: The ICSI-SRI Spring 2005 Speech-to-Text Evaluation System. *In: Rich Transcription 2005 Spring Meeting Recognition Evaluation Workshop*. Edinburgh, UK: Springer LNCS 3869.

Teleki, Csaba, Szabolcs, Velkei, Levente, Tth Szabolcs, & Klra, Vicsi. 2005. Development and evaluation of a Hungarian Broadcast News Database. *In: Forum Acusticum 2005*.

Tranter, S. E., & Reynolds, Douglas A. 2004. Speaker Diarisation for Broadcast News. *In: IEEE Odyssey Speaker and Language Recognition Workshop 2004*.

Tritschler, Alain, & Gopinath, Ramesh A. 1999. Improved Speaker Segmentation and Segments Clustering Using the Bayesian Information Criterion. *In: Eurospeech'99 - 6th European Conference on Speech Communication and Technology*.

Tsai, Wei-Ho, Cheng, Shih-Sian, & Wang, Hsin-Min. 2004. Speaker Clustering of Speech Utterances Using a Voice Characteristic Reference Space. *In: ICSLP 2004 - 8th International Conference on Spoken Language Processing*.

Vandecatseye, An, & Martens, Jean-Pierre. 2003. A Fast, Accurate and Stream-Based Speaker Segmentation and Clustering Algorithm. *In: Interspeech'2003 - 8th European Conference on Speech Communication and Technology*.

Vandecatseye, An, Martens, Jean-Pierre, Neto, Joao, Meinedo, Hugo, Garcia-Mateo, Carmen, Dieguez, Javier, Mihelic, France, Zibert, Janez, Nouza, Jan, David, Petr, Pleva, Matus, Cizmar, Anton, Papageorgiou, Harris, & Alexandris, Christina. 2004. The COST278 pan-European Broadcast News Database. *In: LREC 2004 - 4th International Conference on Language Recourses and Evaluation*.

Varma, Krishnaraj. 2002. *Time-Delay-Estimate Based Direction-of-Arrival Estimation for Speech in Reverberant Environments*. Ph.D. thesis, Virginia Polytechnic Institute and State University.

Viswanathan, Mahesh, Beigi, Homayoon S.M., Dharanipragada, Satya, & Tritschler, Alain. 1999. Retrieval from Spoken Documents Using Content and Speaker Information. *In: Fifth International Conference on Document Analysis and Recognition (ICDAR'99).*

Waheed, K., Weaver, K., & Salam, F. M. 2002. A robust algorithm for detecting speech segments using an entropic contrast. *In: MWSCAS-2002. The 2002 45th Midwest Symposium on Circuits and Systems.*

Webb, Andrew. 2002. *Statistical Pattern Recognition, Second Edition.* Second edition edn. John Wiley & Sons.

Wegmann, S., Zhan, Puming, & Gillick, L. 1999a. Progress in Broadcast News transcription at Dragon Systems. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 1999.*

Wegmann, Steven, Scattone, Francesco, Carp, Ira, Gillick, Larry, Roth, Robert, & Yamron, Jon. 1999b. Dragon Systems' 1997 Broadcast News Transcription System. *In: 1999 DARPA Broadcast News Workshop.*

Wooters, C., & Huijbregts, M. 2007. The ICSI RT07s Speaker Diarization System. *In: Rich Transcription 2007 Meeting Recognition Evaluation Workshop.* Baltimore, MD, USA: Springer LNCS 4625.

Wooters, C., Mirghafori, N., Stolcke, A., Pirinen, T., Bulyko, I., Gelbart, D., Graciarena, M., Otterson, S., Peskin, B., & Ostendorf, M. 2004. The 2004 ICSI-SRI-UW Meeting Recognition System. *In: Rich Transcription 2004 Fall Workshop.*

Wu, T. Y., Lu, L., Chen, K., & Zhang, H.-J. 2003. UBM-Based Real-time Speaker Segmentation for Broadcasting News. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2003.*

Yamamoto, Kiyoshi, Asano, Futoshi, Yamada, Takeshi, & Kitawaki, Nobuhiko. 2006. Detection of Overlapping Speech in Meetings Using Support Vector Machines and Support Vector Regression. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, **E89-A**(8).

Youn, D., Ahmed, N., & Carter, G. 1982. On using the LMS algorithm for time delay estimation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **30**(5).

Zdansky, J., & David, P. 2004. Automatic Audio Segmentation of TV Broadcast News. *In: Radioelektronika 2004*.

Zdansky, Jindrich, David, Petr, & Nouza, Jan. 2004. An Improved Preprocessor for the Automatic Transcription of Broadcast News Audio Stream. *In: ICSLP 2004 - 8th International Conference on Spoken Language Processing*.

Zhang, Tong, & Kuo, C.-C. Jay. 1999. Heuristic approach for generic audio data segmentation and annotation. *In: ACM International Conference on Multimedia 1999*.

Zhou, Bowen, & Hansen, J. H. L. 2005. Efficient audio stream segmentation via the combined T^2 statistic and Bayesian information criterion. *IEEE Transactions on Speech and Audio Processing*, **13**(4).

Zhou, Bowen, & Hansen, John. 2000. Unsupervised Audio Stream Segmentation And Clustering Via The Bayesian Information Criterion. *In: IEEE International Conference on Acoustics, Speech, and Signal Processing 2000*.

Zhu, X., Barras, C., Lamel, L., & Gauvain, J. L. 2007. Multi-Stage Speaker Diarization for Conference and Lecture Meetings. *In: Rich Transcription 2007 Meeting Recognition Evaluation Workshop*. Baltimore, MD, USA: Springer LNCS 4625.

Zhu, Xuan, Barras, Claude, Meignier, Sylvain, & Gauvain, Jean-Luc. 2005. Combining Speaker Identification and BIC for Speaker Diarization. *In: Interspeech'2005 - 9th European Conference on Speech Communication and Technology*.

Zhu, Xuan, Barras, Claude, Lamel, Lori, & Gauvain, Jean-Luc. 2006. Speaker Diarization: From Broadcast News to Lectures. *In: Rich Transcription 2006 Spring Meeting Recognition Evaluation Workshop*. Bethesda, MD, USA: Springer LNCS 4299.

Zibert, Janez, Mihelic, France, Martens, Jean-Pierre, Meinedo, Hugo, Neto, Joao, Docio, Laura, Garcia-Mateo, Carmen, David, Petr, Zdansky, Jindrich, Pleva, Matus, Cizmar, Anton, Zgank, Andrej, Kacic, Zdravko, Teleki, Csaba, & Vicsi, Klara. 2005.

The COST278 Broadcast News Segmentation and Speaker Clustering Evaluation - Overview, Methodology, Systems, Results. *In: Interspeech'2005 - 9th European Conference on Speech Communication and Technology.*