MANCHESTER 1824

The University of Manchester

Speaker Diarization

Project Progress Report

May, 2012

By: **Ubai SANDOUK**

Supervisor: **Dr. Ke CHEN**

The University of Manchester

School of Computer Science

COMP60990 – Research Methods and Professional Skills

Table of Contents

List of Figures

Abstract

Most current voice activated tasks depend on short focused speech or commands from certain people. However, in most environments sound exists continuously. Furthermore, many of the current recordings are long multi speaker ones, such as broad cast news and meetings. It is essential for computing devices to distinguish speech from other sounds; and also, to distinguish one speaker's voice from another and attribute each segment of the conversation to its true speaker. The task mentioned is speaker diarization. Clearly, it takes advantage of all speaker recognition techniques, but often research focus on text-independent recognition. And it benefits many applications, such as transcription and conceptual indexing. This report shows the central preposition of my project: the enhancement of speaker diarization accuracy by the use of a novel speaker specific speech representation.

This report shows a number of widely acceptable approaches for handling the diarization task. Those approaches depend on the underlying representation of the input stream, the most adopted of which is the MFCC. In addition, many of these approaches can be adapted to incorporate a new representation, such as adapting the fixed window metric segmentation for the novel speaker specific representation. Nonetheless, it is necessary to study the novel representation and discover new ways to achieve best accuracy in the diarization task. Empirically, that is done by developing evaluation techniques and performing methodical experiments. Those representations, approaches and evaluation methods are presented in this report. Also, the novel representation is introduced. Towards the end of this report, evidence of the superiority of the new representation is presented by showing some testing results I already performed.

1. Introduction

1.1.Speech

It has become well accepted that people interact with electronic devices via natural language, be it English or otherwise. With the latest technology such as Apple's Siri® (a speech recognition software for the iPhone®) and Microsoft's Kinect® (a gaming device for Xbox360® and windows-based platforms) it seems inevitable for machines to understand the human language. However, for that to become a reality in the future, it is essential to improve the current accuracy of the applications even in the most ordinary tasks, such as deciding who is speaking at the moment or speaker detection.

The human speech is a signal which contains a lot of information including the words, feelings and identity of the speaker. It is up to the receiving party to decode the signal, possibly with different techniques for different pieces of information. In addition, Kemp et al. (2000) emphasises that many recognition tasks depend on being fed short single-speaker speech signals (e.g. the Siri application and the Kinect device); however, a great number of current sound streams are long multi speaker steams; such as meeting and news broadcast data. On the other hand, the real time conduct is one of the most important features of human speech interaction. Therefore; effective, highly accurate and time-efficient methods are necessary to deal with large amounts of speech information.

1.2.Project Scope

This project is meant to handle the speaker diarization task; i.e. the task of determining which parts of a speech stream is uttered by each speaker. The main objective is to improve the current speaker diarization accuracy by investigating appropriate approaches. In particular, using earlier research done by Chen and Salman (2011a;b), leading to the construction of a novel speech representation which holds the greatest speaker discrimination (discussed in 3.5) which is expected to improve the task at hand.

This objective may be further decomposed into a number of smaller goals, including:

- Finding the statistical metric which is most faithful to the speaker identity when using the speaker specific representation.
- Surveying the current state-of-the-art approaches of the speaker diarization. In addition to the empirical evaluation methods used by the research community.
- Using the findings to improve the speaker segmentation and later the speaker clustering tasks.
- The assessment of the improvement achieved by using the speaker specific representation.

These goals already raise some grand research challenges, such as the discriminative power of a certain representation and the underlying model of human speech when using that representation. In addition, they direct research into the understanding of human speech and the sound interpretation in our brains.

The diarization task is thought of as a two-step process: (a) *Speaker Segmentation*: where the speaker change points are detected; and (b) *Speaker Clustering*: where the different speech segments uttered by the same speaker are grouped together. Previous research tries to extract speaker-related information directly from the stream. What is more is that the new representation readily contains the most identity discriminative information. Therefore, a wise use of this representation must improve the overall accuracy.

In this project a number of the issues will be covered, such as the underlying model and how to learn better representations (such as GMM and HMM), which leads to the development of theoretical distance measures to use. In addition, this project is deeply concerned about the separability and clustering of speakers.

The project can be categorized under speech processing or digital signal processing. In addition to speaker segmentation and clustering, the project includes tasks such as speaker comparison and speech activity detection. The full process scheme is presented in section 4.1. Another way of looking into the matter is whether the processing is done online (where only information from previous parts of the stream is available) or offline (where

the information is used regardless of where it is found in the stream). And therefore, online (similarly offline) segmentation and clustering are parts of the project as well.

1.3.Report Structure

In this **Introduction**, the major objective of the project is discussed and the main reasoning behind it is presented.

The **Background** part illustrates the background knowledge needed for the speech processing done in the project. Starting with sound, signals and signal representation; section 2 explains the basics of human speech and speech processing. It also covers one of the fundamental models, the *source/filter model*, and then explains the main differences in speakers' sound and corresponding signals. Section 3 lists the most important features of a speech signal used in any speech related application, such as the frequency envelops and MFCCs. Section 3 also describes the novel speaker specific speech representation, which is then heavily used. Section 4 supplies the background on the current research outlook of speaker diarization; including the metrics and models used for the segmentation and clustering tasks. Section 4 then covers current research in the diarization fielded.

The **Project Progress** part of the report demonstrates what the project involves and points out what has been achieved. Section 5 shows the empirical justification of the validity of the current approaches, including the corpora used and evaluation criteria most adopted by the research community; and then shows the preliminary results of using some diarization techniques. In addition, Section 5 identifies the tools and methods used (or will be used).

2. Signal and Signal Processing

This section describes the background knowledge in speech digital representation and manipulation. It also provides one of the most fundamental models for human speech, the source/filter model. At the end, I briefly describe the sources of differences between speakers.

2.1.Sound

Sound is formed when a medium vibrates; for instance, the vibrations of air molecules or pressure. These vibrations can be modelled as two types of layers interleaved and travelling together through the medium; higher pressure layers (molecules compressed more than normal) and lower pressure layers (molecules compressed less than normal). These vibrations of air affect the ears causing people to hear. In fact, sound can be observed as a signal; the amplitude of which corresponds to the pressure change and the length corresponds to the distance between two high (or two low) pressure layers.

Human speech is one form of sound which people have developed to carry valuable information; such as thoughts and feelings. However, it also carries other inherited characteristics such as the speaker identity.

2.2.Signals

A signal is the continuous measure of a quantity in terms of time. An example of a signal is the measured voltage of a certain point in an electric circuit. A signal that repeats itself every period T is called a *periodic signal*; with the value T being its *period*. The number of times the signal occurs in a time unit is called the *frequency* (mathematically, the inverse of the period). An example of a periodic signal is the sinusoid given in the formula:

$$x(t) = \sin(t)^{1} \tag{2.1}$$

Often signals are additive in the time domain. According to Coleman (2005, p.72) Jean Fourier provided the means where any real world signal can be represented as the sum of a number (maybe infinite) of sinusoids with different frequencies, phases and magnitudes. One common technique is straightforwardly called *Fourier Transformation*; which has only one condition: the original signal must be finite in its length and its amplitude. This transformation generates a new representation of the signal in the *frequency domain*[2]. Where each frequency is associated with a value representing how much the sinusoid of that frequency contributes to the original signal.

It is common to look at the *frequency spectrum* of a signal, especially a speech signal, which is composed of many mixed frequencies (caused by the shape of the human vocal tract). Figure 2.1(a) shows the time spectrum for the sum of two sinusoids (frequencies 100 and 220 hertz, with some artificially added noise). It is very hard to observe the frequencies in the time domain. Figure 2.1(b) shows the result of Fast Fourier Transformation of the signal in (a).

2.3.Sampling

The continuous nature of the sound signal can be captured through analogue devices, such as microphones. However, for the digital processing it is impossible to retain this continuity. The only possible way is to keep measures at distinct times. The process of retrieving those measures at equal intervals is called *sampling.* It is usually carried out by an *analogue-to-digital (AD)* convertor. The time period between two consecutive measures is called the *sampling period* T_s; and the number of samples taken in one second is called the *sampling frequency* F_s. Figure 2.2(a) shows sampling of an analogue signal at *frequency* $F_s = 1/T_s$.

Evidently, the sampling processes may case losses in the original signal. For example, if only one sample was taken each period of a sinusoid, a flat signal would be perceived. However, taking two samples of each period of the previous sinusoid may be enough to encode the whole signal. *Nyquist Frequency* is the highest frequency of a signal that can be faithfully preserved when sampled at a certain frequency F_s; after which the original signal may be distorted, i.e. *spatially aliased*. It can be proven that Nyquist Frequency is in fact, ½F_s (Jurafsky and Martin, 2009).

---

[1] This is the simplest form of a sinusoid. The general form is given: $x(t) = A.\sin(wt + \emptyset)$ where A is called *Amplitude*; w is called *Angular Velocity*; and $\emptyset$ is called *the Phase*.

[2] In practice, the phase component is represented in the complex part of the result. Therefore, it is often discarded and only the frequency component (the real component) is kept.

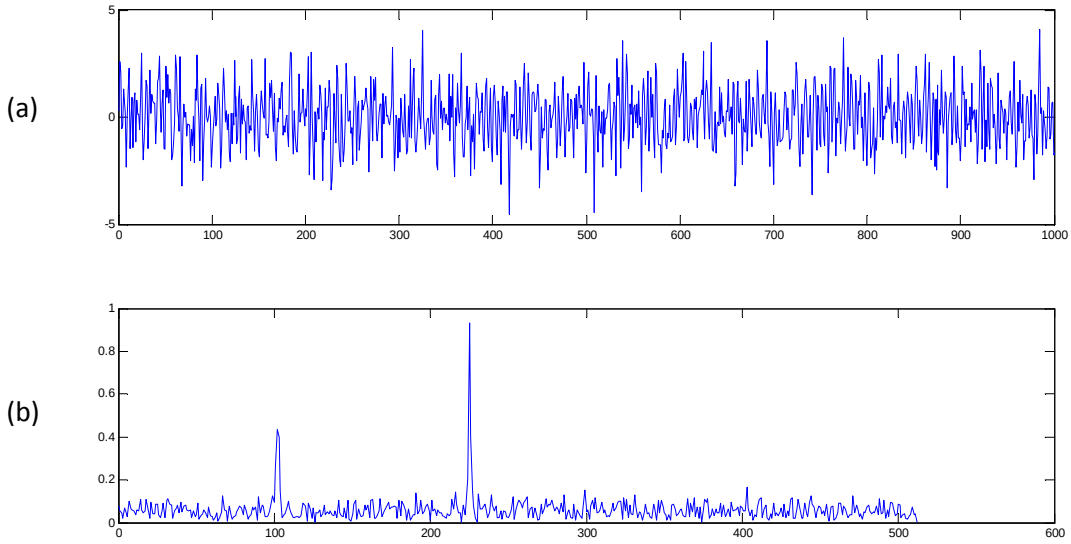


Figure 2.1 :(a) The time spectrum and (b) frequency spectrum of a signal.

2.4.Quantization

The sampling process produces a train of real values $x[t]$; each representing the magnitude of the signal at time nT_s. The domain of these real values is continuous and cannot be represented for a digital machine. Therefore, an approximation of each sample value to one of distinct levels is made; and the value is then stored as the level number in a number variable. The more levels are used the closer the approximated value to the real value (and the smaller the approximation error) will be; but the more space a single sample will consume. Figure 2.2(b) shows the signal sampled in figure 2.2(a) and the result of quantization is shown as a shaded area.

The approximation errors caused by quantization are irreversible. Therefore, it is often the case of using more quantization levels than needed. In practice, it is common to use 16 bits per sample, i.e. 65536 levels. Digitally, manipulating the digital signal is done by manipulating individual samples. For instance, doubling the value of a range of samples would mimic double the volume of the sound of that range.

2.5.Windowing

Window functions are used as a temporally bound on the original signal. The most common window function is the *rectangle function*. The effect of this window on the original signal is to produce time bounded signal that is similar to the original inside the rectangular range and is set to zero outside of it. This helps further processes to attend only to the bounded part. Other window functions preserve more attention to the central values of the window over the boundary ones. Examples include *Hamming*, *Hanning and Triangular*. It is common to use overlapping windows over the same signal. Each window will have an extract of the original signal without any interference among the results. Digitally, windowing is thought of as a function affecting the samples of the signal to produce a new series of samples, i.e. a new signal. Figure 2.3 shows the affects of different window functions on a digital flat signal.

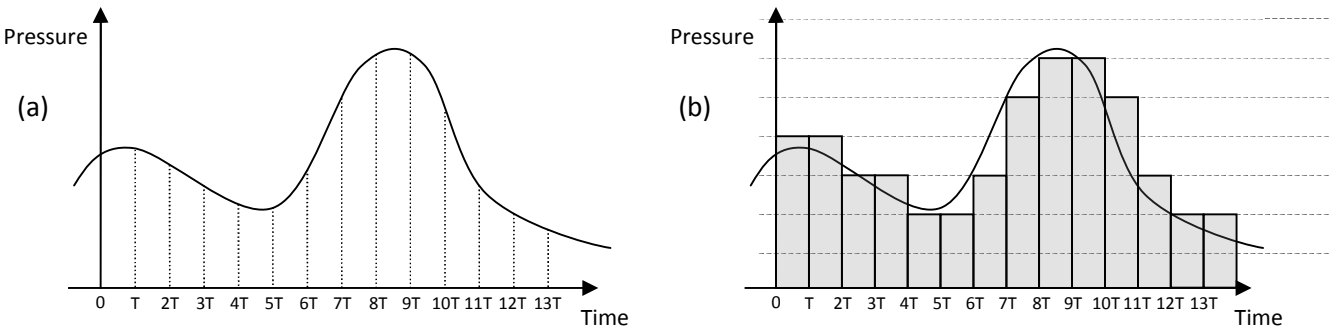


Figure 2.2: (a) Sampling at frequency f=1/T and (b) Quantization a continuous signal.

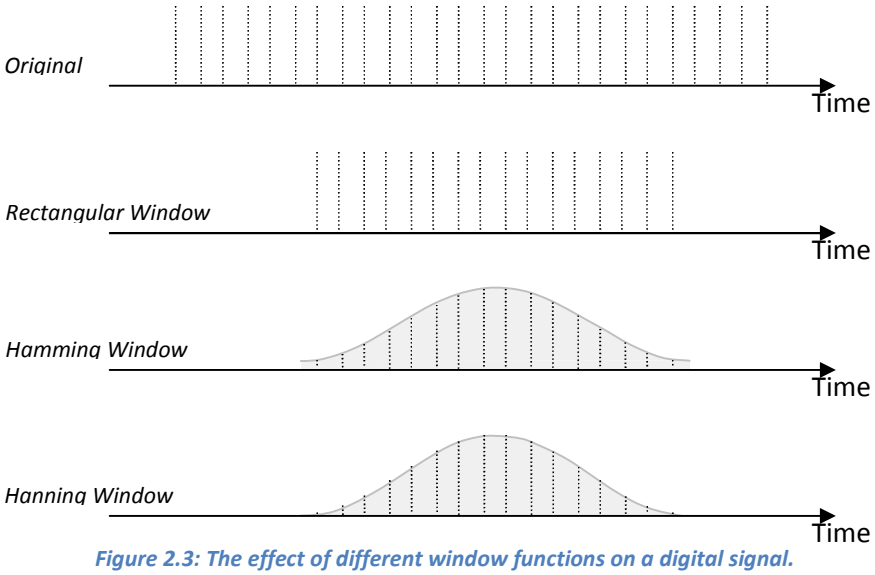


Figure 2.3: The effect of different window functions on a digital signal.

2.6.Filtering

A filter affects the signal in the frequency domain; i.e. it alters the components of the signal at specific frequency ranges. For instance, an ideal low pass filter only keeps low frequency components and clears any component over a certain frequency step (called the *cut-off frequency*). Just like windowing, filtering a signal will result in a new signal with sample values affected by the filter. The simplest way to build a digital filter in practice is by the use of the *difference equation* (Coleman, 2005, pp.54-60); which determines the value of the output sample as a linear combination of the previous input (and/or output) sample values. An example would be the *Averager*; i.e. averaging previous N input samples, which will result in a signal less influenced by the quick changes (or high frequencies) of the original signal, mimicking a low pass filter. Another example is the *Differentiator*; i.e. averaging the difference of the past N input samples, which will result in a signal less influenced by the slow changes (or low frequencies) of the original signal, mimicking a high pass filter. In general the following equation is used:

$$y[t] = b_0x[t] + b_1x[t-1] + b_2x[t-2] + \cdots + b_kx[t-k] \\ +a_1y[t-1] + a_2y[t-2] + \ \ldots + a_jy[t-j] \tag{2.2}$$

Using the different values for the coefficients (a's and b's) results in different filter behaviour. The resulting filter is known as the Butterworth filter. In practice these values are retrieved (or estimated) from lookup tables according to the ratio of the cut-off frequency to the sampling frequency of the input signal. In addition, combining two filters consecutively allows the build of a pass band filter (only passes a certain range of frequencies).

Filters are often described by their impulse response; i.e. the output corresponding to a single *Dirac impulse* $\delta(0)$. On one hand, if the output is finite then the filter is called *Finite Impulse Response (FIR) Filter*; and according to Coleman (2005, p. 60) the corresponding difference equation would only reference the input signal. On the other hand, if the output is infinite in power or in time the filter is called *Infinite Impulse Response (IIR) Filter;* and the corresponding difference equation would reference both the input and output signals.

A *filter bank* is a collection of filters that covers the entire range of frequencies found in the original signal. Each filter outputs a different signal; which can be further processed individually. An example is the sound equalizer, each frequency range is boosted separately; afterwards a final sound signal is produced by merging the results of the boosted signals. More details are presented in Huang, Hon and Reddy (2001, pp. 251-260).

2.7.Human Speech

The *vocal tract* is the part of the human body responsible for producing voice. It mainly consists of three cavities; the *pharynx cavity*, the *mouth cavity* and the *nasal cavity*. Those cavities shape the final voice. As a person exhales, he pushes the air out of the lungs and through the *voice box*, where it either causes the *vocal cords* to vibrate or pass between them directly to the mouth cavity. When the vocal cords vibrate, they vibrate periodically at a base frequency known as the *Fundamental Frequency* (f_0). Sounds that excite the vocal cords are called *voiced* sounds. For example, all the vowels are voiced. On the other hand, if the vocal cords are extra relaxed or extra stiff the air pass

between them and the result sound is said to be *unvoiced*. For example, the sound /f/. These cavities change their shape by the aid of the tongue, teeth, lips and the nasal tissues which effectively change the final sound. More details are presented in (Rabiner and Bing-Hwang, 1993, pp.14-17).

The smallest part of the spoken language is called a *phoneme*. Researchers have made much effort to list all the phonemes a person can articulate; and also to write every word in a certain language phonetically. The most common notation is the *International Phonetic Alphabet (IPA)*. For example, the English word "fish" can be phonetically written as /fɪʃ/. Different phonemes have different characteristics that can be used to recognise them. The most observable feature of phonemes is whether it is voiced or unvoiced, which comes from the existence of the base frequency or not.

The ears are the part of the human body that is affected by sound signals. The air vibrations affect the eardrum, which in turn vibrates and causes other bones to vibrate. The higher the pressure change of the air the louder the perceived sound is. *Pitch* is the perception of the fundamental frequency. The ear can be excited by only a range of frequencies; on average, an adult can hear frequencies up to 20,000 Hz. Therefore, in practice it is common to have sampling frequencies a bit over F_s=44,000 Hz to avoid distortions in audible sounds. Moreover, the ear is more sensitive to certain frequencies than others. The Mel scale captures this phenomenon. It is described in section 3.4.

The frequencies of one's speech change dramatically over time (see 2.8 for more details). It is not wise to do a frequency analysis over a long time of speech because it will produce wrong frequency components. Therefore, a small sliding window (<20 ms) is often used. The result is a 3D diagram (Time – Frequency – Intensity). A special diagram called the *spectrogram* is used. A 2D plan represents the Time-Frequency dimensions, and the colour brightness represents the intensity. An example of spectrogram is presented in Figure 2.4. The window used is 2ms. The speech signal did not exceed 6000 hertz at any given window.

2.8.The Source/Filter Model

The architecture of the vocal tract has influenced a *source/filter model* of human speech. The source is a simple base-frequency signal generator that is used for voiced phonemes. This signal passes through the cavities of the voice tract. Much like any physical tube enclosing a signal; the different cavities' shapes cause a raise of the effect of certain frequencies and a drop of the effect of others. The result voiced sound is different because of the filtering against different frequencies that takes place in the cavities. For example, the difference between the vowels /æ/ in "cat" and /u/ in "boot" is only the shape of the lips. This effect can be modelled using a filter bank applied to the filtering) is a key feature for the detection of voiced phonemes. This is why the speech detection process (both in human and digital systems) relay heavily on voiced phonemes (Rabiner and Bing-Hwang, 1993, p.24).

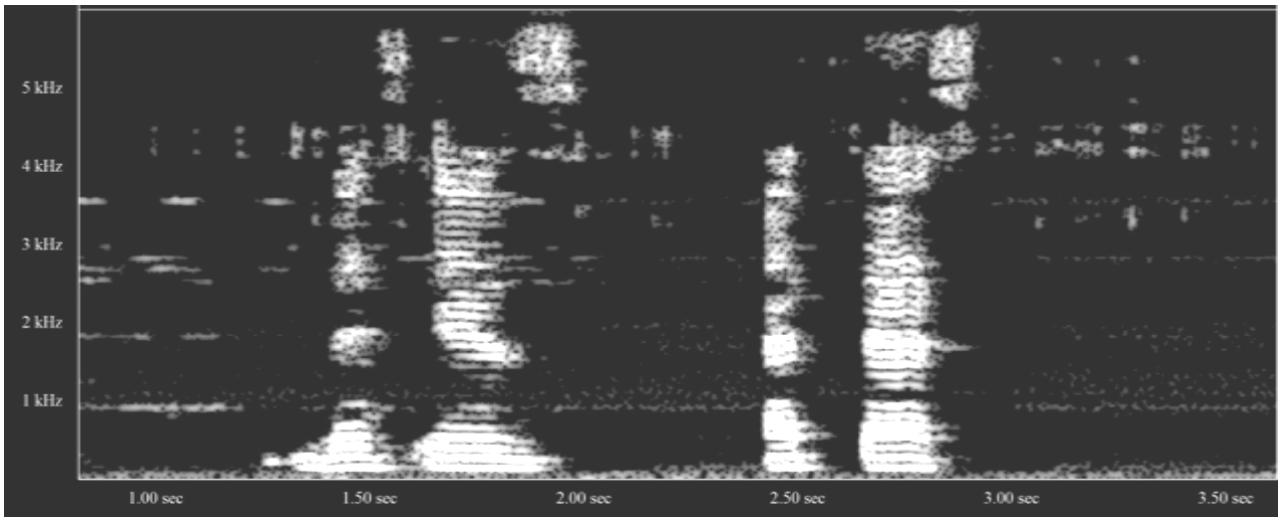


Figure 2.4: Signal Spectrogram of the speech of the sentence "This is a test" (a male speaker)[3].

[3] The figure is generated by the a free tool "*Spectrogram 16*" [online] Available at:<http://www.visualizationsoftware.com/gram> [Accessed 24 March 2012]

This model is the basis for many speech synthesizers such as *Klatt's formant synthesizer*; presented in Klatt (1980) and further discussed and its parameters set in Coleman (2005, pp.62-68). Clearly, different banks are used for different phonemes. Hence, there will be a great number of parameters to control the synthesizer. This resembles the cavities different positions and shapes to form the final voice.

2.9. The Source of Speakers' Differences

The *frequency of the vibrating vocal cords* (the fundamental frequency) is the source of most differences between voices. It can be used for some general classification such as gender (male voices usually have lower frequency than female voices) and age (the fundamental frequency decreases with age). The *full length of the vocal tract* and the ability to change its shape is also a factor of differences between voices.

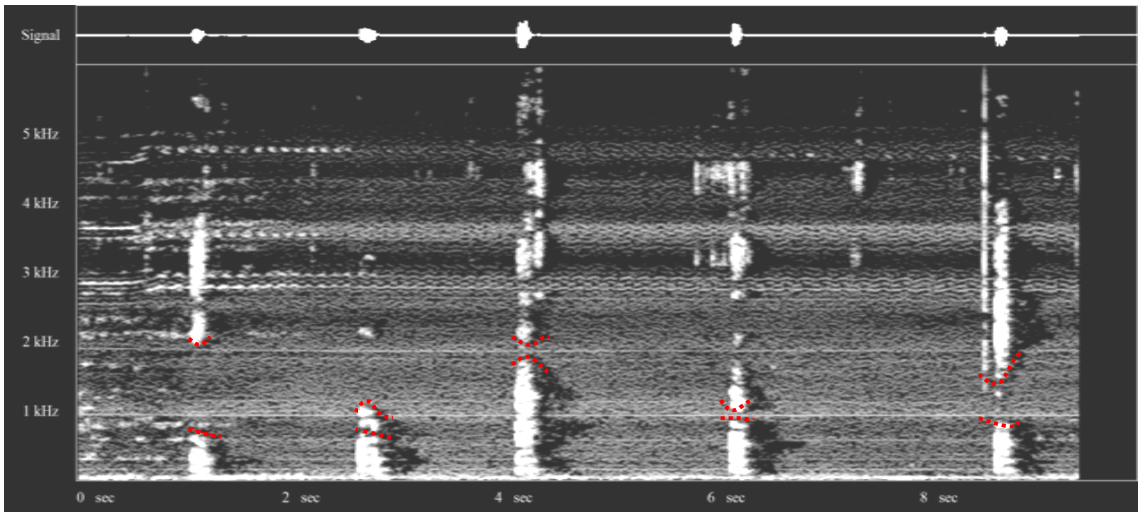
In a phoneme-based study we find differences in *class and dialect* between different people. Those changes affect the places of stress and the syllables used (Campbell, 1997). Also a source of difference is the number of uttered syllables in a period of time or *rate of speech* (Jurafsky and Martin, 2009). That also affects by the speaking style of the person, such as the deletion of the last syllable of a word, the reduction of the stress in some cases and the merge of certain words for convenience (Huang, Hon and Reddy, 2001). Moreover, the same person speech may be affected by their mood, such as repetition, whispers and yells.

3. Speech Signal Features

As seen before the human voice signal is full of information. However, it is required to extract it from the series of sample values available to the digital process. Therefore, first features are extracted from the signal and are handled for certain purposes afterwards. In this section, some generic speech features are presented, followed by the description of the most widely used features for speaker related tasks, i.e. MFCC. Later in this section speaker specific representation is described. Detailed LPC, phoneme features and MPEG-7 features are presented in Appendix A.

3.1. Phoneme Features and Classification

A spectrogram of one voiced phoneme shows the boost of a certain set of frequencies. Figure 3.1 shows 5 vowel phonemes and their respective spectrograms. It is clear that each vowel has a set of boosted frequencies (The colour intense area). The visual boundaries of these frequencies (the frequency of change from damp to boost or the other way around) are called *the formants*. A phoneme can have up to 5 formants called F_1 to F_5. All voiced phonemes share the lower frequencies (around the fundamental frequency F_0); but they differ on higher frequencies. For
is speaker dependant and not flawless (Rabiner and Bing-Hwang, 1993, p.27). More phoneme classification is presented in Appendix A.



Vowel	/i/	/u/	/ə/	/ɔ/	/æ/
Example	Eve	Boot	Ago	Bought	Cat

Figure 3.1: The first two formants of the presented vowels.

3.2.Energy

The sum of the samples' values over a period of time is called the frame's *Energy* (Coleman, 2005, p.48).

$$Energy = \sum_{n=t_1}^{n=t_2} x[n] \tag{3.1}$$

Clearly, the energy increases with the size of the frame. Therefore, it holds no inherited significance by itself. On the other hand, *Power*: the average of energy over the length of the frame, does.

$$Power = \frac{\sum_{n=t_1}^{n=t_2} x[n]}{t_2 - t_1} \tag{3.2}$$

Root Mean Square (RMS) amplitude is a more used measure because it is not affected by the signs of the values.

$$RMS = \sqrt{\frac{\sum_{n=t_1}^{n=t_2} (x[n])^2}{t_2 - t_1}} \tag{3.3}$$

In speech processing, these measures have different uses. For example, a frame with RMS close to zero, suggests that it is a silence frame. Also, a sudden change in the RMS for a short period suggests an irregular event; such as a knock or a phone ring. If the change is long enough it may indicate a speaker change.

The *zero-crossing rate* is the rate of the signal sign change. It can be easily approximated using of the power over small consecutive frames. This feature is of great importance in distinguishing different types of voices, such as different types of music or even human speech (Gouyon and Pachet, 2000).

3.3.Frequency Analysis and Pitch Estimation

The fact that each voiced phoneme has certain frequency features (see 2.8.) suggests carrying on the signal study to the frequency domain; by the use of Fourier Transformation. A good digital implementation for the discrete version is *Fast Fourier Transformation* FFT, which can be made to frames of length 2^N; if the length is not a power of 2, the frame is zero padded. An inversion function is also available, called *Inverted FFT.* Since the results of mentioned transformations are in the complex numbers domain $\mathbb{C}$, it is often the case that applying the Inverse FFT to the frequency spectrum of a signal will not retrieve the same original signal.

Typically, voiced phonemes hold more energy in low frequencies range than unvoiced phonemes; where the energy is spread all over the spectrum (Coleman, 2005, p.86). This fact can be used to estimate voiced/unvoiced regions. Noticeably, this method needs a threshold to distinguish between voiced and unvoiced regions. Another better way of voiced region estimation is *autocorrelation* where a window of the signal is auto-correlated with itself at growing intervals. The interval with the highest autocorrelation is said to be the base period and can be used to determine the base frequency F_0. If the highest autocorrelation was at period 0 then the frame is unvoiced. Parameters for this method include the lower and upper bounds for the autocorrelation process; for human voice, good values are 40 hertz and 600 hertz respectively (Huang, Acero, Hon and Reddy 2001, p.325).
Figure 3.2(A) shows the time spectrum of the /o/ sound of the word "boy" for 90ms sampled at 62K hertz. It clearly shows the voiced nature of the vowel phoneme. The frequency spectrum is presented in 3.2(B) up to 4096 hertz.

3.4.Cepstrum/MFCC

As described in Coleman (2005, p.78) the frequency spectrum (such as the one in Figure 3.2(B)) contains two types of oscillations. 1) The small more frequent ones; called *harmonics*. Their frequency is of a multiple of the base frequency. Therefore, they are related to the speaker. 2) The large hidden ones, where the oscillations are amplified; their frequency is affected by the other formants of the speech. For a speaker related tasks, one may be interested in the rate of the harmonics' repeat; i.e. the frequency of them; therefore, FFT is applied on the frequency spectrum to analyze its oscillations. The result is in the *samples' domain*, known as the *Cepstrum domain*. The x axis represents samples called *quefrency.* The amplitude reflects the amount by which the quefrency value contributes to the frequency spectrum. The quefrency which has the highest amplitude and is reasonably high in value corresponds to the harmonics frequency and also to the base frequency of the original signal[4].

[4] There is no guarantee that this is F_0. However, it is in relation with F_0.

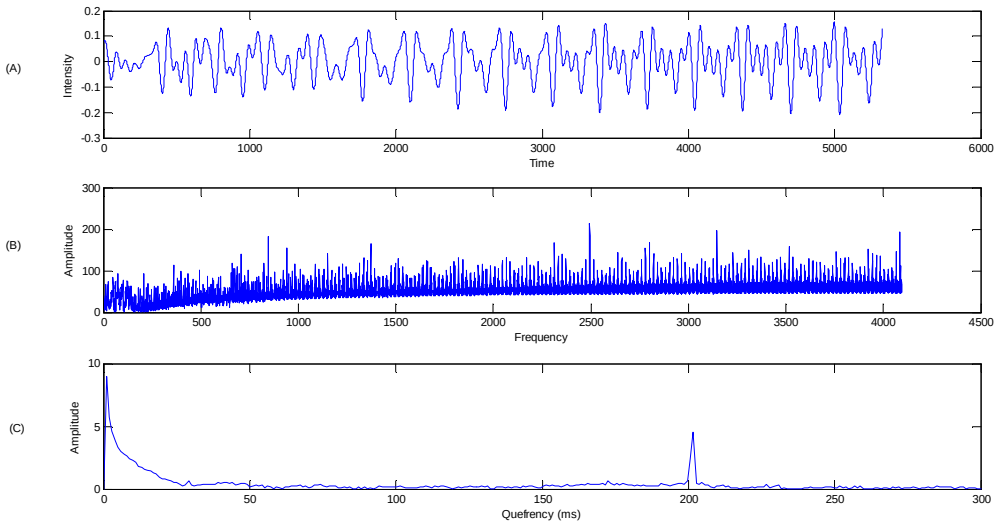


Figure 3.2: (A) The time, (B) frequency and (C) quefrency spectrums for the sound /o/ of the word "boy"

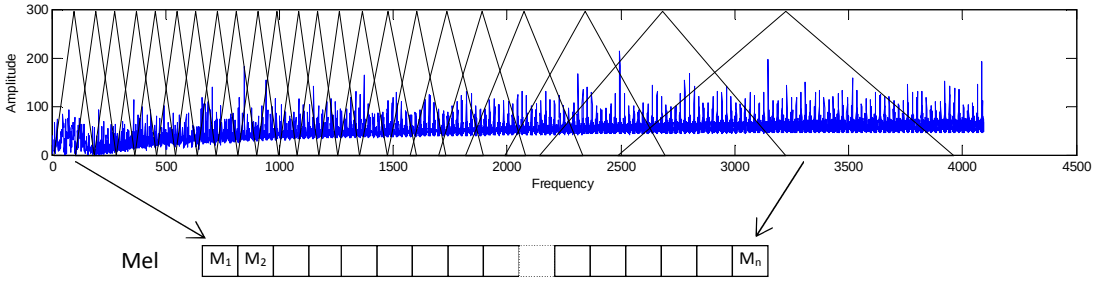


Figure 3.3: the Mel overlapping triangular filter bank

Directed by a field testing of human perception of different frequency; the *Mel-Scale* handles frequencies below 1000Hz with a linear behaviour; and frequencies over a 1000Hz logarithmically; hence giving more importance to lower frequency changes. Handling signals on the Mel-Scale can be made by the use of a filter bank with equal length regions below 1000Hz and log length regions afterwards. Often, the filters used are 20 overlapping triangular filters. To produce MFCCs those filters add up the energy within (from the frequency spectrum); each will produce a single value called the *Mel value*. The logarithms of the Mel values are introduced to the FFT again (more formally to the *Discrete Cosine Transformation* DCT which is the FFT in $\mathbb{R}$). The result is called *Mel-Frequency Cepstral Coefficients MFCCs*. Figure 3.3 shows the Mel-Scale step. The MFCCs reflects the quefrency behaviour and therefore encodes speaker related information. They reduce the affects of the source of the original signal; and keep only the affects of the filter (as in the source/filter model). Moreover, the Mel-Scale step provides more significance to the lower frequencies, and hence conveys the fact that humans are more influenced by changes at lower frequencies than changes at higher ones (Jurafsky and Martin, 2009).

The MFCCs have provided good results in both speaker and speech recognition tasks. Often only the leading L numbers are kept for each frame, such as 12 or 20. Also, The *delta MFCCs* are the difference between MFCCs of two consecutive frames. Moreover, the *delta-delta MFCCs* are the difference between delta MFCCs of two consecutive frames. The deltas have also shown good performance in the recognition tasks.

3.5.Speaker Specific Features

A novel approach for representing speech retaining as much speaker discrimination information as possible is presented by Chen and Salman (2011a and 2011b). A *deep neural architecture* (DNA) is used to learn essential features of the speaker. The DNA is trained with labelled MFCC frames; with the objective of grouping frames uttered by the same speaker and discriminating frames uttered by different speakers; without the loss of the fundamental objective in any DNA which is the minimization of the difference between the input and the output. One of the hidden layers is called the *code layer*; contains two types of values: speaker related and speaker non-related values; the latter are used in the reconstruction process. To be able to learn to discriminate properly, two networks are propagated simultaneously with and the difference in the code layer is minimized if they belong to the same speaker or maximized otherwise. This multi-objective loss function can be summarized by the following

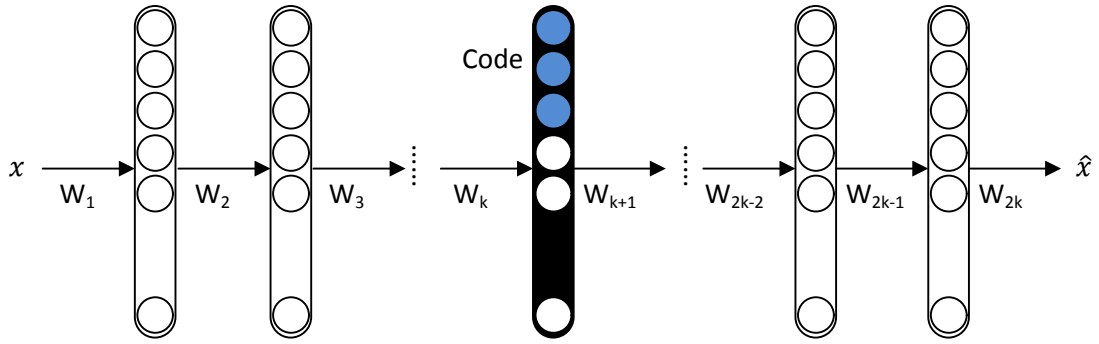


Figure 3.4: a DNA. The shaded layer is the code layer. The shaded neurons are the speaker specific features.

($0 < \alpha < 1$; controls the trade off between the reconstruction loss and the discrimination loss):

$$\mathcal{L}(X_1, X_2; \Theta) = \alpha[\mathcal{L}_R(X_1; \Theta) + \mathcal{L}_R(X_2; \Theta)] + (1 - \alpha)\mathcal{L}_D(X_1, X_2; \Theta) \quad (3.5)$$

$\mathcal{L}_R$ is the reconstruction error and can be straight forwardly calculated by:

$$\mathcal{L}_R(X_1; \Theta) = \frac{1}{T_B}\sum_{t=1}^{t=T_B}\|x_t - \hat{x}_t\|_2^2 \quad (3.6)$$

However, $\mathcal{L}_D$ is the discrimination loss. $\mathcal{L}_D$must be small for same speaker and large for different speakers. The writers propose the following as the compatibility measure between two frames.

$$D(X1, X2; \Theta) = \|\mu_1 - \mu_2\|_2^2 + \|\Sigma_1 - \Sigma_2\|_F^2 = D_m + D_S \quad (3.7)$$

Where μ is the mean value of the code vector; and Σ is its covariance matrix. $\|.\|_2$ is the $\mathcal{L}_2$ norm and $\|.\|_F$ is Frobenius norm (i.e. $\|A\|_F = \sqrt{tr(AA^*)}$). Therefore the discrimination loss can be expressed as:

$$\mathcal{L}_D = \mathrm{I}\, D(X1, X2; \Theta) + (1 - \mathrm{I})(e^{-\frac{D_m}{\lambda_m}} + e^{-\frac{D_S}{\lambda_S}}) \quad (3.8)$$

$\mathrm{I} = 1$ if the frames are taken from the same speaker, and $\mathrm{I} = 0$ otherwise. λ_m, λ_S are boundary parameters that can be estimated from the training set. According to the writers, not having a reconstruction loss will result in the DNA over-fitting the training data.

The speaker specific representation of the frame is obtained from the code layer. 100 features are extracted for each MFCC frame. Using the MFCC representation removes the effects of the source and generates more robust performance for the recognition task. However, using MFCC it is unclear how to determine which frames are similar enough to be deemed from the same speaker. The DNA is trained to have the representation of the frames form the same speaker close and from different speakers far. Therefore, developing more effective recognition is possible.
Figure 3.4 is a reproduced figure of the DNA presented originally in (Chen and Salman, 2011b, p.2), the code layer is dark shaded and the speaker specific features are also shaded in the code layer.

3.6.Summary

A number of sound features were presented in this section; ending with the speaker specific representation. The features presented are generic, i.e. they work in any speaker related task, such as speaker identification or speaker diarization. Clearly, each representation has advantages and disadvantages. Moreover, each needs individual further processing techniques. For the purpose of this project, a comparison will be done to show the superiority of the DNA representation over other representations in terms of accuracy. Nonetheless, the higher dimensionality of the DNA (100 features for each frame) may cause some issues and be time consuming.

4. Speaker Diarization

Speaker diarization is the task that answers "*who spoke when?*". In this section, the task of speaker diarization is introduced in details, along with the most common approaches to handle it. Depending on the number of channels of the same recording available, single and multi channel diarization are presented. Next the current methods for speaker segmentation are listed. Later the current methods for speaker clustering are also listed. A mention of the current systems is provided towards the end.

4.1.Single Channel Diarization

Speaker diarization partitions the input stream into parts; each uttered by a single speaker and then determines the regions spoken by each individual speaker (Kotti et al., 2008). Several applications benefit from this process; such as dialogue detection, automatic annotation of broadcast news and automatic indexing of speech according to speaker. It is common to think of the diarization task as "blind"; with no prior knowledge about the content of the stream (Tranter, 2006). However, this is not a constraint and some research do benefit from this knowledge. Furthermore, the quality of the input stream may be a source of errors; such as: low signal to noise ratio, low bandwidth or other human artefacts such as heavy breathing (Almpanidis and Kotropoulos, 2008).

Speaker diarization is thought of as a two-step task (Kotti et al., 2008):

1) *Speaker segmentation*: the task of determining *turn points* in a stream. That is where the change of the speaker occurs. The output is a stream of segments, each uttered by a single speaker, and no two adjacent segments uttered by the same speaker.
2) *Speaker clustering*: the task of assigning each segment to its speaker, so that it can be determined what parts was spoken by each speaker.

A classification of diarization systems can be made by approach (Anguera et al., 2012):

1) *Bottom-Up*: the algorithm is initialized be a very large number of clusters, each representing a speaker. When two clusters are determined to belong to the same speaker, they are combined. An extreme example would be the frame to cluster approach; where each frame is initially thought of as being from a different speaker.
2) *Top-Down*: the model is initialized by a few clusters, each representing a speaker. Whenever a cluster is determined to include more than one speaker it is split into two. An example would be the E-HMM approach, where the model is initialized by only one HMM node (representing the only cluster) and nodes are added until the HMM faithfully represent the stream (see 4.7.).

4.2.Multi Channel Diarization

Depending on the scenario and the application; the diarization system may be provided with a single stream of sound to diarize, or a number of streams for the same recording. An example of multi-channel recording is a recording of a meeting that has many microphones in the meeting room. More information may be extracted from the multi-channel recordings; Anguera et al. (2006) use the multi channels to build an enhanced single reference speech signal using a technique called *Beam Forming*. Beam forming uses a sliding window and the information of relative delay of each microphone from one elected reference microphone, and then aligns the different channels into one. Boakye et al. (2008) use the multi channels for overlapping separation. This separation helps in eliminating the effects of non speech activity (such as door knocking or paper shuffling) on the stream and extracting more accurate features of the recording. Anliker et al. (2006) use the time delay to estimate the speaker position and use that information to improve speaker segmentation. A single channel diarization box flowchart is presented in Figure 4.1 and a multi-channel diarization box flowchart is presented in figure 4.2. It is clear that multi channel diarization is the same as single channel diarization in its core, except it uses multi channel features; which are not always available. That is why I will focus only on single channel diarization.

4.3.Voice Activity Detection

Voice activity detection (VAD) is used to separate speech parts from silence (or any non-speech) parts. Clearly the accuracy of VAD depends on the speech clarity and bandwidth of the channel. During a clear conversation, a silence frame will have zero energy. The intuitive VAD process is by measuring each frame's energy. However, this does not detect other kinds of non speech; such as, music and laughter. Other methods such as GMM were suggested (Tranter, 2006). In (Wooters et al., 2004) a supervised method is used with a three state HMM trained to detect speech/non-speech/music. In (Sinha et al., 2005) VAD was used for the bandwidth detection and gender recognition using universal gender models (see 4.8.) which improves the clustering accuracy. In (Liu and Kubala, 1999) 8 classes differentiation was made: 3 of speech and 5 for non-speech; the writers claim that 80% of the true speaker change points happen in non-speech frames and therefore it is essential to get VAD correct. The obvious drawback of any supervised method proposed is the need for external data to train the model. Some research tries to handle non-

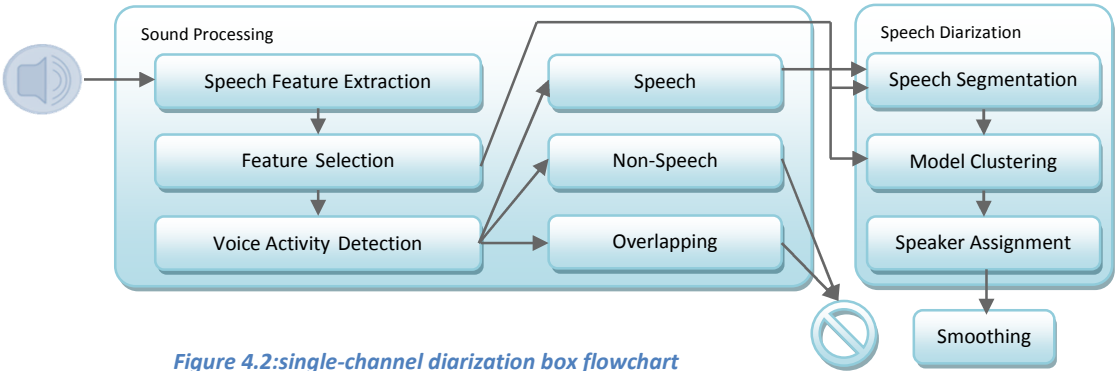


Figure 4.2:single-channel diarization box flowchart

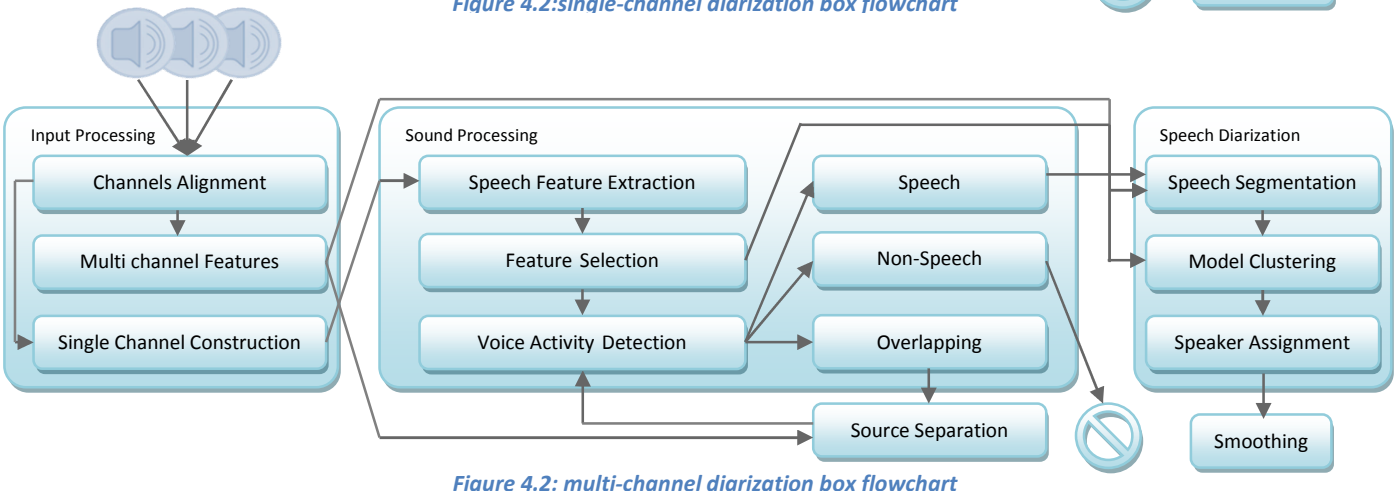


Figure 4.2: multi-channel diarization box flowchart

speech as an extra speaker, and therefore eliminates the need for an explicit VAD process. However, experiments have shown that having a separate VAD improves the overall performance (Anguera et al., 2012).

4.4.Speaker Segmentation

Most approaches for finding *turn points* in a multi speaker stream can be explicitly categorized as.

- *Metric based*: often the assumption that one speaker's speech follows a normal distribution is made. Hence; statistical metrics, called *divergences* (measuring how far two normal distributions are), can be used to check if two speech segments are uttered by the same speaker or not. The method by which this idea is implied can be further subcategorised into two methods (Cheng and Wang, 2010).
    - *Fixed size window*: Two fixed sized windows move frame by frame through the stream. The dissimilarity is measured between them; which results in a distance curve over the entire stream. This curve can be further processed to find peaks or major dissimilarity points. The main metrics used are listed in 4.5. Kotti et al. (2006) provides a comparison between three systems and concludes that different measures put emphasis on different evaluation criteria such as reducing the false alarm rate of reducing the miss rate. However, some of the best performing systems need manual parameter setting.
    - *Growing window*: A window is expected to contain one turn point or otherwise it will grow by one frame. Once the window is decided to have a turn point the process is repeated starting from the turn point. The metric used to make the decision of the most probable change point is usually a Bayesian Information Criterion (BIC) or any of its variations. This process is evaluated in (Cettolo et al., 2005). Clearly, the process is very computationally demanding. Therefore, some research has been done to pre evaluate regions and find probability of turn points before applying the BIC metric (see 4.8).
- *Model based*: A model is trained in a supervised fashion for each speaker. These models are then used to estimate turn points. According to Kotti et al., (2008) the most common model based method is GMM/UBM presented in (Reynolds, 2000). It relays on the Gaussian Mixture Model (GMM) to represent one speaker; which is a multimodal Gaussian probability distribution. That is to say that the feature vector x representing a sample of speech from a speaker λ follows the following distribution:

$$p(x|\lambda) = \sum_{i=1}^{M} w_i p_i(x) \quad (4.1)$$

The weights w_i are learnt for each speaker in the training phase through an Expectation Maximization (EM) algorithm. However, this is one part of the solution; the other part is calculating the probability of the feature vector x not be associated with speaker λ i.e. $p(x|\bar{\lambda})$. To do so, one could test the vector x for every other speaker in the training set. This process, other than being time consuming, cannot cope when an unanticipated speaker is encountered. Therefore, a general GMM model for all human speech, Universal Background Model (UBM), is developed during the training process. The more training data available the better the models would be. Lastly, a classifier is built using an error-minimization approach. And the stream frames can then be tested against the model.

Other models are rarely used such as *anchor models* (Collet et al., 2003) where the speaker is not represented as an absolute form like in the GMM, but rather as a vector of its likelihood to a certain set of predefined speakers. This vector is called the speaker characterization vector (SCV) and their differences can be measured using L_p norm metrics[5]. (Anguera and Bonastre, 2010) builds on top of that a faster model called *binary keys*. It has shown comparable performance to anchor models at much less time consumption.

4.5.Distance Measures

A number of distance measures can be applied to sliding fixed-sized windows P and Q. In fact any divergence metric could be used. More statistic work is presented in (Bimbot et al., 1995); however, the following is a list of the closed forms of the most used distance measures according to kotti et al., (2008): μ is the mean vector of the features; Σ is the covariance matrix of the features and n is the number of frames in the window. tr is the trace operator.

- *Kullback divergence*

$$D_{KL2}(P \parallel Q) = \frac{1}{2}(\mu_Q - \mu_P)^T({\Sigma_P}^{-1} + {\Sigma_Q}^{-1})(\mu_Q - \mu_P) + \frac{1}{2}tr({\Sigma_P}^{-1}\Sigma_Q + {\Sigma_Q}^{-1}\Sigma_P - 2I) \quad (4.2)$$

- *Bhattacharyya divergence*

$$D_B(P \parallel Q) = \frac{1}{4}(\mu_Q - \mu_P)^T(\Sigma_P + \Sigma_Q)^{-1}(\mu_Q - \mu_P) + \frac{1}{2}log\frac{|\Sigma_P + \Sigma_Q|}{2\sqrt{|\Sigma_P\Sigma_Q|}} \quad (4.3)$$

- *Log Generalized Likelihood Ratio (GLR) divergence*

$$\Delta D_{GLR}(P \parallel Q) = \frac{n}{2}\ log|\Sigma| - \frac{n_P}{2}\ log|\Sigma_P| - \frac{n_Q}{2}\ log|\Sigma_Q| \quad (4.4)$$

- *Information based: Rényi divergence*

$$D_{EL}(P \parallel Q) = R_\alpha(P \parallel Q) = \frac{1}{2}(\mu_Q - \mu_P)^T(\alpha\Sigma_P + (1-\alpha)\Sigma_Q)^{-1}(\mu_Q - \mu_P) - \frac{1}{2\alpha(\alpha-1)}log\frac{|\alpha\Sigma_P + (1-\alpha)\Sigma_Q|}{|\Sigma_P|^{1-\alpha}|\Sigma_Q|^{\alpha}} \quad (4.5)$$

Rényi divergence is generic with a parameter α. Usually α is considered in the range [0,1] but that is not a restriction of the metric. When $\alpha = 1$ the metric converges to Shannon entropy $E(P - Q)$. Interestingly, when $\alpha = 0.5$ the equation will be identical to the Bhattacharyya divergence.

Chen and Gopalakrishnan (1998) formulated the speaker change problem as a model selection problem; where either two speakers contributed to the speech in the window; and therefore two normal distributions fit the window best. Or else only one speaker contributed to the speech in the window; and therefore only one normal distribution fits it best. That formulation made the use of the ΔBIC measure accepted. This is currently used to select among the following two hypotheses:

$H_0 : (x_1, \dots, x_N) \sim N(\mu, \Sigma)$ no speaker change during the whole window.

$H_1 : (x_1, \dots, x_i) \sim N(\mu_P, \Sigma_P) \text{and} (x_{i+1}, \dots, x_N) \sim N(\mu_Q, \Sigma_Q)$: a speaker change takes place in the middle of the window. The selection usually requires a threshold which is set manually, or set to zero and controlled by the λ parameter. And the ΔBIC is given in the following closed formula:

$$\Delta BIC(P \parallel Q) = \frac{n}{2}\ log|\Sigma| - \frac{n_P}{2}\ log|\Sigma_P| - \frac{n_Q}{2}\ log|\Sigma_Q| - \frac{1}{2}\lambda\left(d + \frac{1}{2}d(d+1)\right)logn = D_{GLR}(P \parallel Q) - \lambda.P \quad (4.6)$$

d is the number of features considered; and λ is a human set parameter. P is a penalty term to control the complexity of the selected model (e.g. favour a simpler model).

[5] $L_q(V) = \sqrt[q]{\sum|V(i)|^q}$

4.6.Speaker Clustering

The purpose of the clustering is to join all the segments uttered by one user in one unity. Preferably the system would produce one cluster for each speaker. When the speaker is modelled as a single modal normal distribution, distance can be measured through any of the metrics presented for segmentation such as the KL divergence. Otherwise, such as using GMM models, more complex techniques are used such as the use of a HMM (a node for each GMM and a learnt transition function) with an alignment algorithm. Nonetheless a problem in clustering is the deciding on the correct number of clusters. Two families of clustering techniques are described:

- *Offline*: starts after segmentation and uses all the segments to determine the final clustering. An example is hierarchical bottom-up clustering presented in (Tranter, 2006); where a number of iteration takes place. Each iteration, the two closest clusters are merged. This process is repeated until the closest two clusters are far enough (i.e. surpass a manual threshold) to describe two distinct speakers and should not be merged.
- *Online*: may start even if the segmentation has not finished yet, uses information of previous segments only. (Lilt and Kubala, 2004) presents and compare three methods: Leader-Follower Clustering (LFC), Dispersion-based Speaker Clustering (DSC) and Hybrid Speaker Clustering (HSC). In LDC, one cluster is initialized for the first segment. Later, for each segment a comparison is done with all the previous clusters; if the segment is close enough to one of the clusters, the cluster is updated to include that segment. Otherwise, a new cluster is initialized by the new segment and added to the clusters pool. In DSC, the same is carried out except for the desicion of spawning a cluster; which is done according to a Likelihood Criteria $G(c)$ rather a dissimilarity threshold given in eq 4.7. Penalizing large covariance and large number of clusters.

$$G(c) = \left|\sum_{j=1}^{j=c} N_j \Sigma_j\right| \sqrt{c} \tag{4.7}$$

  Where c is the number of clusters and N_j is the number of frames in the j[th] cluster. $G(c)$ is computed in two cases, adding the segment to the closest cluster or spawning its own cluster; and the minimum is used.

When a frame-to-cluster approach is carried out; it might be the case that some frames are mistakenly assigned to a cluster that differs from the ones all around it. Further, the boundary frames of a turn point may be misclassified. Therefore, there is a need for a cluster purification scheme.

4.7.One Step Diarization

It is clear that the same metrics are used in both the segmentation and in the clustering tasks. Besides, the accuracy of the clustering is enormously affected by the accuracy of the segmentation. Plus, the accuracy of the segmentation can be improved by clustering (detection of a missed turn point or a removal of a hypothesised turn point). Thus, a one step segmentation and clustering approach may be carried out. According to Anguera et al., (2012) either a bottom up; or top down. A speaker is usually modelled by a GMM and the transitions by a HMM. Then a dynamic algorithm aligns the frames into the clusters; such as the Evolutive HMM approach (Meignier et al., 2000).This approach has the advantage that the information gathered from the clustering process can improve the segmentation. However, it is extremely computationally demanding.

4.8.Overview of Current Diarization Systems

The metric segmentation family members differ in the use of features and metrics. They often focus on the use of one metric with a tuneable threshold to control the ratio of miss rate and false alarms. The most adapted metric is the BIC. In Chen and Gopalakrishnan (1998), BIC is used to find turn points. First, it is used to find at most a single turn point in a window; and then adapted to find many turn points. However, it is shown that the application of BIC is very computationally demanding. Therefore, many approaches have been proposed to make the whole process faster. For instance, the DISTBIC approach; proposed by Delacourt and Wellekens (2000). DISTBIC is a two-phase segmentation approach. In the first phase a low miss rate metric is used to quickly over segment the stream. This is usually done with a fixed size window approach. In the second phase the ΔBIC is applied to assess each turn point and remove the ones separating two consecutive segments uttered by the same speaker. Many have tried to improve this approach; such as, Lu and Zhang (2002) using KL2 before the BIC decision and Zhou and Hansen (2005) by using T^2 metric in the first phase, both succeeding in reducing the time consumption without much sacrifice in accuracy.

The problem with this research direction is that it makes the assumption that a single speaker speech follows a single modal Gaussian distribution; therefore the statistical measures would work. Another direction uses *multi modal Gaussian distribution* to model a speaker (GMM) as is used in (Gangadharaiah et al., 2004) for two speaker conversations; where GLR is using in the first phase to generate candidate turn points and GMMs are trained for both speakers and then used to refine the overall segmentation.

In the model segmentation family a general model for human speech; the Universal Background Model (UBM); is often trained and it is used for speech/non-speech separation even for unknown speakers in a supervised environment. Such a model and a corresponding distance measure are used by Wu et al. in (2003). A universal gender model (UGM) can also be exploited if required (Kotti at al., 2008). Other features can be extracted from the corpus and used in the diarization task, such as the probability of change from one speaker to another (such as in a news broadcast environment). In (Meignier et al., 2000) a HMM structure is incorporated; each speaker is represented by a node and the likelihood of change from one to another is learnt from the training data.

In the speaker clustering task, the major problem is selecting the correct number of speakers. Whether applying a top-down or a bottom-up approach the algorithm can measure the distance between two instances (speakers) by a statistical metric and it continues until the correct number of speakers is achieved. Siegler et al. (1997) use KL2 as a metric and set an acceptable threshold distance as a stopping criterion. The obvious limitation is the need for a threshold set a priori. In (Tritschler and Gopinath 1999) a BIC based metric is used as stopping criteria; which is considered better because of the amount of research done to eliminate the parameters from BIC. The mentioned approaches make the assumption that a single speaker's utterances follow a single modal Gaussian distribution; which is always not the case. Jin et al. (2004) use GLR as a metric for GMM modelled speakers; which fit their selected features better than a single modal Gaussian distribution. In addition, In (Meignier et al., 2006) each speaker is modelled by a GMM and the whole stream is modelled by a HMM. A top down approach is used in building the HMM; and then the stream is aligned to the new HMM model. No more nodes are added when no more gain is achieved. At which point the system stops. It is clear that using a gender discriminator (such as UGM) or a bandwidth discriminator would help increase the cluster purity.

Other interesting techniques have also shown effective in improving the diarization performance. For instance, Tranter (2005) uses more than one system's output and then uses a judge to determine the correct diarization. In Sturim et al. (2011) nine different systems are fused together to perform the diarization task using a cross entropy criterion.

4.9.Summary

In summary, the diarization can be viewed as a two step process, Segmentation and Clustering. Each has been studied extensively. Metric-based and model-based approaches have been proposed to solve both problems and surveyed in this section. It is yet unclear how to use the DNA representation to improve the results of these tasks. Chen and Salman(2011a) proposes a modified KL metric to do metric segmentation. In their experiments the DNA representation provided better results than its MFCC counterpart.

5. Project Progress

In this section I present the experiments' settings to be later used in the empirical study.

5.1.Standard Corpora

In the context of speech processing the *Linguistic Data Consortium* (LDC) provides a number of linguistic datasets, including *TIMIT* and *NTIMT*. TIMIT has 630 speakers (438 males and 192 females) of different backgrounds and dialects. Every speaker uttering 10 English sentences in a quiet environment. NTIMIT is obtained by passing TIMIT sentences through different telephone lines. For our purposes, a random set of those utterances are coalesced to produce a mixed speech which mimics the characteristics of a normal conversation in terms of speaker change. Moreover, they can be artificially overlapped and Gaussian noise can be added to imitate ordinary human speech.
Other less known corpora are sometimes also sometimes used; such as CNET in French and KING/NKING, which are similar to TIMIT/NTIMIT respectively. However, I will limit my experiments to the TIMIT/NTIMIT corpora.

5.2.Evaluation Criteria

The **Voice Activity Detector** (VAD) is evaluated by the number of frames it miss-categorizes. Usually, with *miss rate* (VAD suggests a silence frame for a speech frame) and *false alarm* (VAD suggests a speech frame for a silence frame) ratios. If the VAD is parameterised (for instance, with an energy threshold) then *Receiver operating characteristic* (ROC) curves are used to show the effect of the parameter. The miss rate and false alarm ratios are graphed against each other for different parameter values.
The **Speaker Segmentation** (SS) task is also evaluated by false alarms and miss segmentation ratios. In addition, ROC curves are used in a parameterized setting. *Equal Error Rate* (EER) is the value of the threshold where the false alarm rate and the miss rate are equal (Campbell, 1997).
The **Speaker Clustering** task can be evaluated by the following measures (Kotti et al. 2008):

- *Cluster Purity*: the measure of how restricted to one speaker each cluster is.
- *Cluster Coverage*: the measure of how restricted to one cluster each speaker is.

In the **Speaker Diarization** one wishes to have a one-to-one speaker/frame relation. Three kinds of errors arise from this mapping:

- *Miss*: when the system suggests a silence for a spoken frame.
- *False alarm*: when the system suggests a speaker for a silence frame.
- *Speaker error*: when the system suggests a different speaker other than the true speaker of the frame.

The final diarization error (DER) is thought of as a combination of the previous three errors' values (Tranter, 2006). Usually the speaker error is penalized by the number of error frames as to give little significance to miss classifying shorter speeches.
Another method of evaluation of a diarization system is by precision and recall (Kotti et al., 2006)

- *Precision (PREC)*: the ratio of correct associations out of all associations made.
- *Recall (RECL)*: the ratio of the correct associations out of the associations that should have been made.
- *F measure (F)*: a one value fusion of PREC and RECL given by $F_\beta = (1+\beta^2).\frac{PREC.RECL}{(\beta^2.PREC)+RECL}$. The most widely used F measure is F_1.

In the NIST RT'09 (NIST, 2009) the error rate used is presented in the following:

$$\sum_{S\in segs}\left\{dur(S).\left(Max\left(N_{Ref}(S),N_{Sys}(S)\right)-N_{Correct}(S)\right)\right\} \quad (5.1)$$

Normalized by the true segmentation error $\sum_{S\in segs}\{dur(S).N_{Ref}(S)\}$. Where $dur(S)$ is the duration of the segment; $N_{Ref}(S)$ is the number of reference speakers in the segment; $N_{Sys}(S)$ is the number of hypothesised speakers in the segment and $N_{Correct}(S)$ is the number of hypothesised speakers whom are true speakers as well. Another metric also give different weights to different speakers.

5.3.Speaker Recognition Evaluation

SRE is an evaluation of current speaker recognition technology conducted by the National Institute of Standards and Technology's (NIST). The SRE has been taking place almost once every two years for the past 16 years. According to

(NIST SRE'12) evaluation plan, the main objectives of the evaluation are: a) Investigation of new proposals in speaker recognition field; b) designing efficient methods to incorporate named proposals; and c) measuring the performance of those methods. My project fits perfectly with those objectives, for it aims to evaluate the novel speech representation in text-independent speaker diarization task; which is categorized under speaker recognition. Therefore, the findings of this project will be included in a participation in the NIST SRE'12.

Several evaluation tasks are proposed in SRE'12. The core task can be summarized by the following steps:

- The system is offered 1000 speakers' data in form of raw speech data to train on. Some of the data offered is passed through a phone and some are interview data. One speaker's data includes one or more sentences.
- The system is expected to build a model for each speaker.
- The system is offered a speech segment x and a hypothesized speaker identity λ. The system's task is to validate the hypotheses and emit a confidence value (defined by $\log\left(\frac{p(x|\lambda)}{p(x|-\lambda)}\right)$) of the validity of the hypotheses.

The performance is evaluated on a number of test segments (i.e. 100,000 in the core test) and a summarized measure is calculated for the whole system.
SRE is an opportunity for researchers to put their research findings into a real test. Many participate for that reason.

5.4.Aims of the project

As described in section 1.2.; the project aims to improve the diarization task by using the newly introduced speaker specific speech representation (Chen and Salman, 2011a; b). The project entails the assessment of the DNA representation in tasks such as speaker segmentation, clustering and voice activity detection; which can only be done empirically. Therefore it must undergo a series of tests to compare the performance of state of the art speech representations (i.e. MFCC) to the DNA representation. For example, in the metric speech segmentation task it is straight forward to change the observations' vectors at each frame from MFCC to DNA and apply the same statistical metrics. Nevertheless, the metrics must be adapted to the DNA representation, or at least their parameters be tuned to provide the best performance. Figure 5.1 shows the resulting ROC curve of applying fixed size window KL and GLR metrics (see section 4.5. for details) on a stream of speech (randomly selected from the TIMIT dataset) comparing MFCCs to DNA representation. Figure 5.1(A) uses 1.2 sec window and Figure 5.1(B) uses 150 sec window. Different points on the curve represent different threshold values.

During the preliminary tests I have already conducted (presented in 5.5.); a number of research challenges arose. Namely:

- The underlying model governing the representation of used: It is well known that one person's speech cannot be modelled by the normal distribution in the time domain. Moreover, MFCCs has been shown to not follow a Normal distribution as well, and therefore most statistical analysis methods of the speech are estimates. More recently, Almpanidis and Kotropoulos (2008) showed that the MFCCs may be better represented with the Generalized Gamma distribution. Nonetheless, as seen in section 4.4., GMMs are heavily used as an

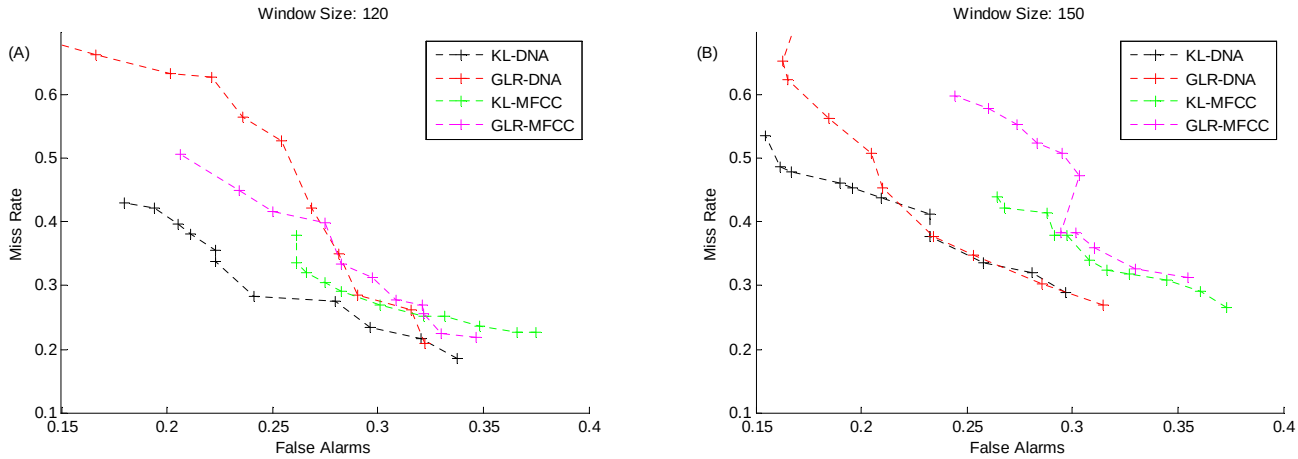


Figure 5.1:The resulting ROC curve of applying fixed window size metric a stream in both MFCC and DNA representation.

underlying statistical model for MFCC features. Nevertheless, the DNA representation is acquired by learning discriminatively using a loss function that combines the first and second order statistics. Therefore, it might be safe to assume that the DNA representation follows a Normal distribution better than MFCCs; and that statistical methods must fit this representation better with no need to use GMMs or any general speaker model.

- The separability of the speaker segments. Even though the DNA achieved good results already in the segmentation task, and therefore it can be used to find changes of speakers in a speech steam. It has shown that two separate speakers can be easily mistakenly clustered together. This effect is further investigated and linked back to the learning process of the DNA and my claim is that it can be solved by a careful study of the DNA features of the stream speakers.
- The stability of the DNA representation amid information loss; such as the distortions and information losses that occurs when filtering the input stream (e.g. phone conversations), and their affects on the DNA representation.
- The affects of the high dimensionality used (100 features per frame) on time and accuracy of application. And also the study of the ability to reduce the number of feature by feature selecting techniques.

In summary, there is a strong experiential evidence for the superiority of the DNA representation in the text-independent speech related tasks. With that in mind, this project is aimed to do the following:

- Study in detail the available methods to incorporate the DNA representation in the segmentation task.
- Present comparative results between the current state of the art speech representation (i.e. MFCC) and the DNA representation in the segmentation task.
- Study the methods of clustering speakers; and techniques to overcome the errors in the segmentation process when using DNA.

5.5.Preliminary Results

5.5.1.Datasets

So far, two datasets are used; namely: *TIMIT* and *NTIMIT*. A number of speakers are selected and then a number of their sentences are coalesced to create a dataset. I keep the original change points and speaker identity as a reference for evaluation. The difference between the datasets is in the bandwidth, and therefore the amount of information contained in the streams. TIMIT is clear while NTIMIT is phone speech with added noise.

5.5.2.System Flow

The system for evaluating the **segmentation** process is presented in Figure 5.2. The system starts by opening the file and removing the silence parts using an energy based approach. Then the DNA representation is extracted (or any representation that is needed). In metric based systems, a window is slid across the representation stream and a distance signal is generated; which later is filtered keeping only the low frequency changes. Speaker changes are expected to be on the local maxima of the distance signal, but only if the maximum satisfies enough change (surpasses a threshold) on both sides (see Figure 5.3). In model based systems, the window is used to classify its content as one of the speakers. After that a list of hypothesised turn points is evaluated.

The model used in all the preliminary tests is a Normal distribution $\mathcal{N}(\mu, \Sigma)$ for each speaker. This claim is made because the deep neural network is discriminatively learning the first and second order statistics. Therefore, it is safe to assume that the DNA representation follows the Normal distribution.

A variation to this flow is the *Anchors model*: a number of hyper-speakers are selected and each frame is represented by its distance to each of those hyper-speakers. Hence, using 32 speakers will result in a 32-features stream. The distance measure used is the Mahalanobis distance:

$$F(x|\mu, \Sigma) = (x - \mu)^T \Sigma (x - \mu)$$

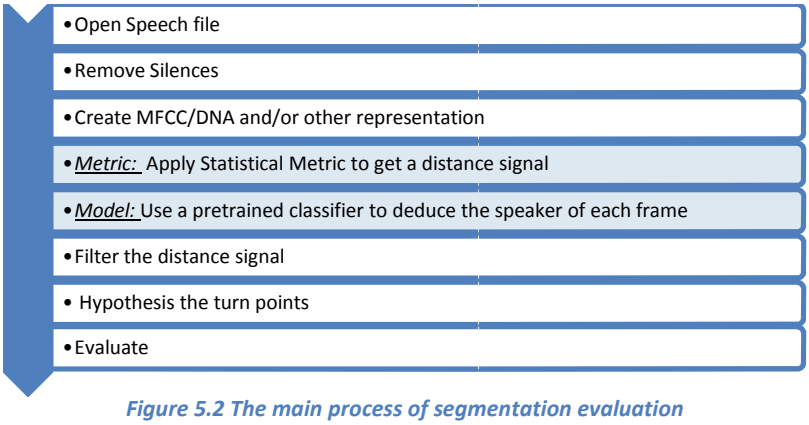


Figure 5.2 The main process of segmentation evaluation

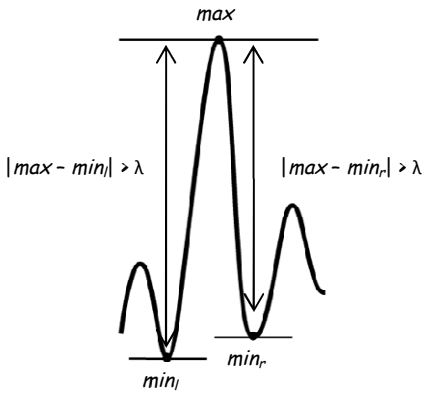


Figure 5.3: The threshold role in analyzing the distance signal

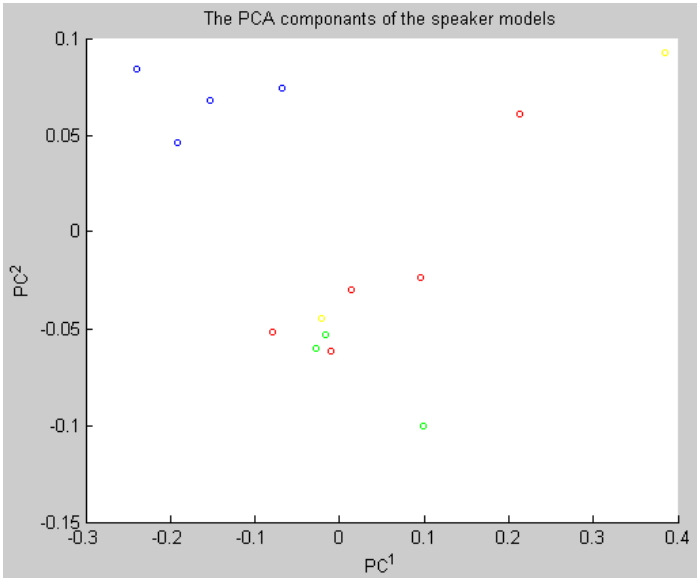


Figure 5.4: The speaker models' first and second PCA.

These processes involve correct choice of a low pass filter and its parameters. This task must be done manually because it is related to the nature of the recording. Moreover, the slight distortions caused by the filtering step are causing segmentation accuracy deterioration.

The dataset has to be extended with new sentences of the each speaker to build individual models (and some never used to build a universal speaker model) before a model based classifier is used. The best results I got were by following this process:

- A fixed size window of 120 frames is slid across the stream with a 30 frames shift.
- For every window, a comparison to all the speakers is made and the closest model is voted the speaker of that window.
- I keep only the difference between each two consecutive windows. That turns out to be the turn point signal.
- To improve the stability, the previous signal is low-pass filtered and further processed much like in the metric approach.

For **clustering**, online clustering techniques might be used. However, so far I could not get proper results because the speaker models are not linearly separable. In Figure 5.4. I show a number of speaker models' first and second principle values. Moreover, they are very close to each other. What I suggest is a deeper study of the DNA features to be able to do the clustering efficiently.

5.5.3. Results

The segmentation is evaluated using ROC curves. The EER line is the equal error rate. The KL modified is a modified KL metric that drops parts concerning only covariance as suggested in (Chen and Salman, 2011a; b).

The full results are listed in Appendix C. According to them, the best segmentation results achieved so far are by a metric approach using the modified KL divergence.

5.6.Research Methodology

The project is planned on two main stages:

- *Background Study*: During this stage, a survey the current state-of-the-art approaches for speaker diarization is completed, including signal processing and speech representation background. Also, the evaluation set is prepared, such as the metrics and the corpora. In addition, I familiarize myself with the Matlab environment the toolboxes used. At the end of this stage, the background report is delivered.
- *The Development*: In this stage the DNA is fully used for the speaker diarization task. Using the evaluation method and datasets from the previous stage, the full findings are presented. At the end of this stage the dissertation is delivered. In which, a detailed review and findings are presented.

The time plan (in Gantt format[6]) is presented in Appendix B. The milestones are the deliverables of the project: a) the Preliminary Report; b) the Background Report; and c) the Dissertation. The shaded navy colour shows the progress in each task (or group) so far. A 20 days period is presented in orange describing the exam period, in which no progress in the project will be made.

5.7.Project Tools

Matlab

Mathworks' **Mat**rix **Lab**oratory (Matlab) is a fourth generation programming tool primarily targeting numerical computations. Through the notion of *Toolbox*es; Matlab's functionality is extended to include almost any computational application. For our certain purposes, Matlab is used to process signals and manipulate statistical data. The most used toolboxes in this project are:

- *Signal Processing Toolbox*: Provided by Mathworks. The toolbox offers functionality to manipulate and visualize digital signals; including filtering and FFT calculation.
- *Statistics Toolbox*: Provided by Mathworks. The toolbox offers algorithms for analysing and modelling data, including multidimensional analysis. The toolbox handles common techniques such as statistical distribution and hypothesis testing (Chi-squared and t-testing).
- *VoiceBox*[7]: Provided by Mike Brookes under GNU Public License. The toolbox offers functionality to deal with speech signals, such as feature extraction, speech recognition and synthesis. For example, the extraction of the MFCC and LPC features.

Conclusion

In recent years, MFCCs have gained enormous attention because of their relation to the human auditory system and the great performance they provide in speaker related tasks. However, it seems that improving the results of the task is not possible with MFCCs anymore. Therefore, researchers are introducing new representations, such as the speaker specific speech representation (Chen and Salman, 2011); models, such as the binary keys (Anguera and Bonastre, 2010); and novel techniques, such as the fusion introduced in MIT's SRE'10 system (Sturim et al., 2011), to improve the diarization task.

It is necessary to understand that any representation must be coupled with a set of techniques and methods to show its true performance towards a certain task. Evaluation methods presented here, including DER and NIST RT'09 equation (eq. 5.1), are acceptable by the research community and often used to evaluate diarization systems. Accordingly, to evaluate the performance of the new representation a number of methods must be implemented and their results compared.

[6] The plan was compiler by the free online tool "gantto" [online] Available on <http://gantto.com> [Accessed 28 April 2012]

[7] Available online <http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html> [Accessed 28 April 2012].

References

Almpanidis, G. and Kotropoulos, C., 2008. Phonemic segmentation using the generalised Gamma distribution and small sample Bayesian information criterion. *Speech Communication*, 50(1), pp.38–55.

Anguera, X., Wooters, C., Peskin, B. and Aguiló, M., 2006. Robust speaker segmentation for meetings: The ICSI-SRI spring 2005 diarization system. *Machine Learning for Multimodal Interaction*, pp.402–414.

Anguera, X. and Bonastre, J., 2010. A novel speaker binary key derived from anchor models. In *Eleventh Annual Conference of the International Speech Communication Association*. pp. 2118-2121.

Anguera, X. et al., 2012. Speaker Diarization: A Review of Recent Research. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(2), pp.356–370.

Anliker, U., Randall, J. and Tröster, G., 2006. Speaker separation and tracking system. *EURASIP journal on applied signal processing*, 2006, pp.1-14

Bimbot, F., Magrin-Chagnolleau, I. and Mathan, L., 1995. Second-order statistical measures for text-independent identification. *Speech Communication*, 17(1-2), pp.177–192.

Boakye, K., Trueba-Hornero, B., Vinyals, O. and Friedland, G., 2008. Overlapped speech detection for improved speaker diarization in multiparty meetings. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*. IEEE, pp. 4353–4356.

Campbell Jr., J.P., 1997. Speaker recognition: A tutorial. *Proceedings of the IEEE*, 85(9), pp.1437–1462.

Cettolo, M., Vescovi, M. and Rizzi, R., 2005. Evaluation of BIC-based algorithms for audio segmentation. *Computer Speech and Language*, 19(2), pp.147-170.

Chen, K. and Salman, A., 2011a. Learning speaker-specific characteristics with a deep neural architecture. *IEEE transactions on neural networks*, 22(11), pp.1744-56.

Chen, K. and Salman, A., 2011b. Extracting Speaker-Specific Information with a Regularized Siamese Deep Network. In *Advances in Neural Information Processing Systems* 25 (NIPS'11). pp. 1-9.

Chen, S. and Gopalakrishnan, P., 1998. Speaker, environment and channel change detection and clustering via the Bayesian information criterion. *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, pp.127–132.

Cheng, S. and Wang, H., 2010. BIC-Based Speaker Segmentation Using Divide-and-Conquer Strategies With Application to Speaker Diarization. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(1), pp.141-157.

Coleman, J., 2005. *Introducing speech and language processing*. Cambridge, UK: Cambridge University Press.

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

Gangadharaiah, R., Narayanaswamy, B. and Balakrishnan, N., 2004. A novel method for two-speaker segmentation. In *Eighth International Conference on Spoken Language Processing*. Jeju Island, Korea, pp. 2337-2340.

Gouyon, F. and Pachet, F., 2000. On the use of zero-crossing rate for an application of classification of percussive sounds. In *Third Conference on Digital Audio Effects*. Verona (Italy), pp. 3-8.

Hermansky, H., 1990. Perceptual linear predictive (PLP) analysis of speech. *The Journal of the Acoustical Society of America*, 87(4), pp.1738-52.

Huang, X., Acero, A. and Hon, H., 2001. *Spoken Language Processing*. New York: Prentice Hall.

International Standards Office, 2001, *ISO 15938-4:2001 MPEG-7: Multimedia Content Description Interface, Part 4: Audio*. [online] Available at: <http://505606.pbworks.com/f/ISO-IEC-15938-4-Audio.pdf> [Accessed 22 April 2012].

Jin, Q. and Schultz, T., 2004. Speaker segmentation and clustering in meetings. In *8th International Conference on Spoken Language Processing*. Jeju Island, Korea, pp. 597-600.

Jurafsky, D. and Martin, J., 2009. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* 2nd ed. New Jersey: Pearson Prentice Hall/Pearson education international.

Kemp, T., Schmidt, M., Westphal, M. and Waibel, A., 2000. Strategies for automatic segmentation of audio data. In *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing.* Istanbul: IEEE, pp. 1423-1426.

Klatt, D., 1980. Software for a cascade/parallel formant synthesizer. *Journal of the Acoustical Society of America*, 67(3), pp.971-995.

Kotti, M. et al., 2006. Automatic speaker segmentation using multiple features and distance measures: A comparison of three approaches. In *IEEE International Conference on Multimedia and Expo*. IEEE, pp. 1101–1104.

Kotti, M., Moschou, V. and Kotropoulos, C., 2008. Speaker segmentation and clustering. *Signal Processing*, 88(5), pp.1091–1124.

Lilt, D. and Kubala, F., 2004. Online speaker clustering. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Montreal, Quebec, Canada: IEEE, pp. I-333-6.

Linguistic Data Consortium (LDC), Philadelphia, PA [Online]. Available: <http://www.ldc.upenn.edu.com>

Liu, D. and Kubala, F., 1999. Fast speaker change detection for broadcast news transcription and indexing. In *Sixth European Conference on Speech Communication and Technology*. Budapest, Hungary, pp. 1031-1034.

Lu, L. and Zhang, H., 2002. Speaker Change Detection and Tracking in Real-Time News Broadcasting Analysis. In *Proceedings of the tenth ACM international conference on Multimedia*. pp. 602-610.

Mathworks, 2012. *Matlab, The Language of Technical Computing* [online] Available at: <http://www.mathworks.co.uk/products/matlab> [Accessed 28 April 2012].

Meignier, S., Bonastre, J., Fredouille, C. and Merlin, T., 2000. Evolutive HMM for multi-speaker tracking system. In *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*. IEEE, p. II1201–II1204.

National Institute of Standards and Technology (NIST), 2009. *The 2009 (RT-09) Rich Transcription Meeting Recognition Evaluation Plan*.[pdf] Available at: <http://www.itl.nist.gov/iad/mig/tests/rt/2009/docs/rt09-meeting-eval-plan-v2.pdf> [Accessed 30 April 2012]

National Institute of Standards and Technology (NIST), 2012. *2012 NIST Speaker Recognition Evaluation* [online] Available at: <http://www.nist.gov/itl/iad/mig/sre12.cfm> [Accessed 28 April 2012].

Rabiner, L. and Bing-Hwang, J., 1993. *Fundamentals of Speech Recognition*. New Jersey: Prentice-Hall.

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

Siegler, M., Jain, U., Raj, B. and Stern, R., 1997. Automatic segmentation, classification and clustering of broadcast news audio. In *Proceedings of the Speech Recognition Workshop*. pp. 97-99.

Sinha, R., Tranter, S., Gales, M. and Woodland, P., 2005. The Cambridge University March 2005 speaker diarisation system. In *Ninth European Conference on Speech Communication and Technology*. Lisbon, Portugal, pp. 2437-2440.

Sturim, D. et al., 2011. The MIT LL 2010 speaker recognition evaluation system: Scalable language-independent speaker recognition. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 5272-5275.

The International Phonetic Association, 2005. [online] Available at: <http://www.langsci.ucl.ac.uk/ipa/> [Accessed 29 April 2012]

Tranter, S., 2005. Two-way cluster voting to improve speaker diarisation performance. In *2005 International Conference on Acoustics, Speech, and Signal Processing*. Philadelphia, PA, pp. 753-756.

Tranter, S., 2006. An overview of automatic speaker diarization systems. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(5), pp.1557-1565.

Tritschler, A. and Gopinath, R., 1999. Improved speaker segmentation and segments clustering using the Bayesian information criterion. In *Sixth European Conference on Speech Communication and Technology*. Budapest, Hungary, pp. 679-682.

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

Wu, T., Lu, L., Chen, K. and Zhang, H., 2003. Universal background models for real-time speaker change detection. In *International Conference on Multimedia Modeling*, pp. 135–149.

Zhou, B. and Hansen, J., 2005. Efficient audio stream segmentation via the combined T^2 statistic and Bayesian information criterion. *Speech and Audio Processing, IEEE Transactions on*, 13(4), pp.467–474.

Appendixes

Appendix A: More Sound Features

In this Appendix I list the less important features of the sound signal for this report as they will not be used. In a more linguistic approach the phonemes are classified in details (see A.2); this helps speech synthesis. Also, some research has been using the readily extracted MPEG-7 features.

A.1. Liner Predictive Coefficients (LPC)

An approximation of the speech signal sample can be made by a linear combination of the previous p values:

$$\tilde{x}[n] = \sum_{i=1}^{p} a_i . x[n-p+i] \quad (3.4)$$

Moreover, if the residuals (i.e. the error $e[n] = x[n] - \tilde{x}[n]$) are saved then the full input signal can be retrieved by first computing $\tilde{x}[n]$ and then adding the residual. This fact has motivated a compression technique for speech signals known as *Liner Predictive Coding* (LPC). The error value is stored for each frame of the signal; as well as the coefficients a_i. The compression is achieved because the error signal amplitude is smaller than the original signal amplitude. However, if p is chosen small then the error values become large and the compression fails. On the other hand, if p is chosen large then the required space for the coefficients will be considerably large and the compression also fails. Either way, the coefficients a_i are best selected to minimise the mean squared error value for the approximation. To do so, the original signal is auto-correlated with itself to generate an auto correlation matrix, of which the coefficients can be extracted. The method is described in details in Rabiner and Bing-Hwang (1993, pp.97-106) and in Huang, Acero, Hon and Reddy (2001, pp.290-300).

A simple observation of the LPC values suggests that the LPC values do not change dramatically unless a sudden event occurs, such as change in the frequency of the speech. Using Fourier Transformation on the error signal generates the *Line Spectral Frequency* (LSF) spectrum; which can be used to estimate the changes in the base frequency and formants of the original speech. It can be shown that they are more stable than other methods for F_0 estimation (Huang, Acero, Hon and Reddy, 2001, p.305). Another use of the LPC is linked to the source/filter model; where the errors can be viewed as the distortion happening for a signal. Either signal (the source and the filter) can be linearly approximated by the previous p values of the original signal. This approximation generates a list of pairs of values (source/filter pairs) that can be further used for analysis of the signal. The list of pairs is called *Line Spectral Pairs* (LSP). Another interesting idea is *Perceptual Linear Predictive* (PLP) presented in Hermansky (1990). PLP performs the Mel-Scale step on the input signal so to boost the effect of lower frequencies before extracting the LSF or LSP.

A.2. More Phoneme Classification

Other than the voiced/unvoiced feature and the formants of the voiced phonemes (see 3.1.), phonemes uttered completely from the nasal tract, such as /ŋ/ in "king" /kɪŋ/, are called *Nasal*. Whereas, Phonemes uttered completely from the mouth tract, such as /t/ in "tip", are called *Plosive*. Phonemes that cause friction in either the mouth or the nasal cavities are called *Fricative*; for example, /ʃ/ in "fish" /fɪʃ/. Fricatives can be voiced (/v/ in "voice") or unvoiced (/ϴ/ in "teeth"). Another case is when the vocal cords are momentarily closed; causing air pressure to build up behind them and then is released at once causing a sound. These sounds are called *Stops*. Stops can be voiced (/g/ in "get") or unvoiced (/k/ in "kit"). Depending on what happens to the vocal cords after the air escapes. Other phonemes are harder to classify, such as the sound /w/ in "wit". They are called *Semivowels* if they are close in sound to some vowel. Other than that they occupy their own category such as the only whisper phoneme /h/ as in "hard". This classification is further refined in (Rabiner and Bing-Hwang, 1993, pp.21-37) and (Huang, Acero, Hon and Reddy 2001, pp.39-47). It is also refined and linked to the signal theory in (Campbell, 1997).

According to Jurafsky and Martin (2009); other features of the voiced phonemes may help the synthesis of speech, such as the position of the tongue (High/Low, Front/Back) and the shape of the lips (Round/Flat).

A.3. MPEG-7 Features

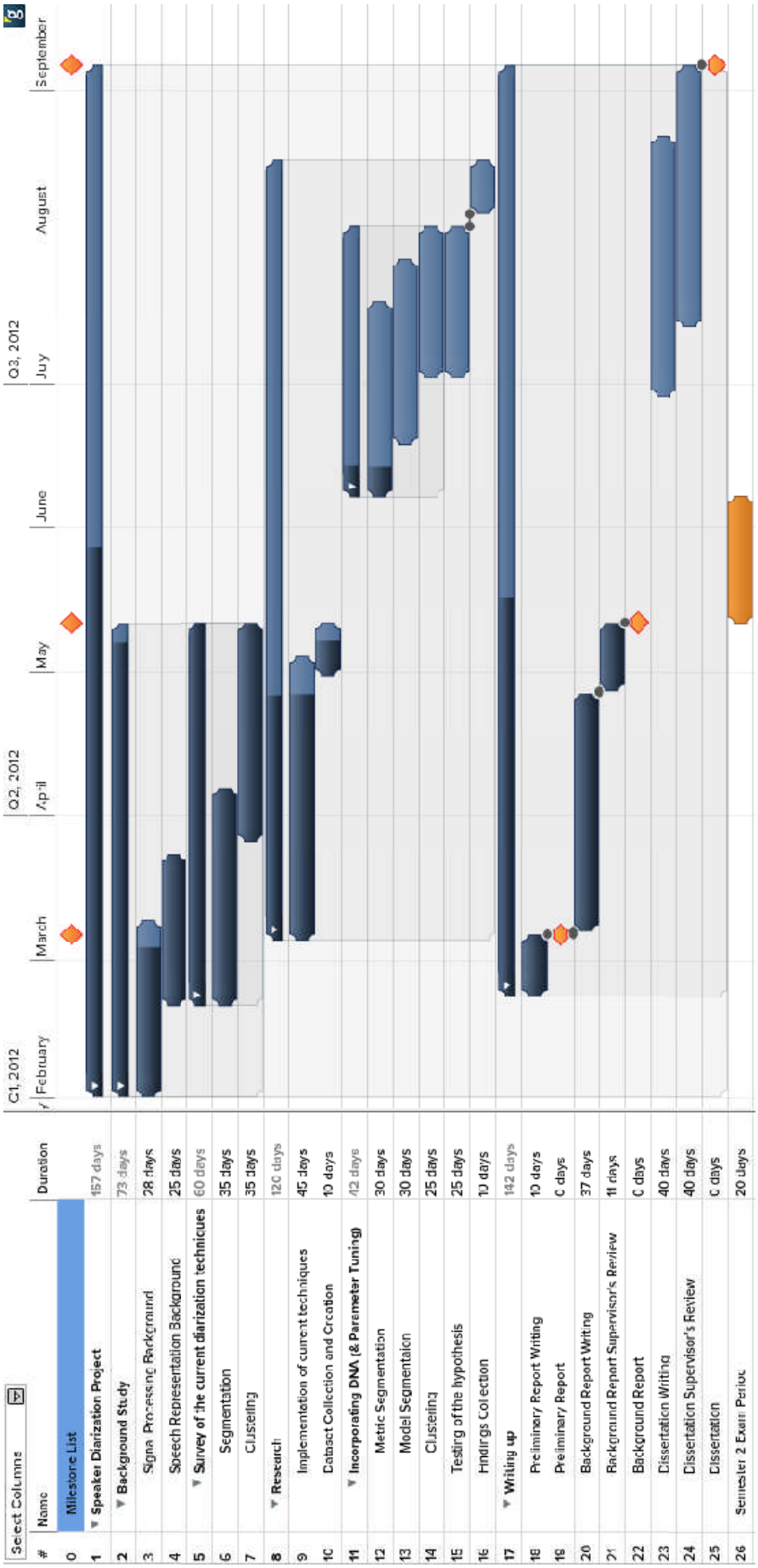
This standard is developed by the Moving Picture Experts Group and is standardized by the International Standards Office in (ISO 15938-4:2001). It encodes many multimedia types including audio. The standard provides the structure of the media file, along with some low-level features embedded in the file or easily extracted from the data. Also, it describes some high-level functionality for application use.

Features are many; I present the most used ones in their defined categories:

- *Basic*: the structure lists the values of the samples.
  The *AudioWaveform* describes the minimum and maximum values of the time spectrum.
  The *AudioPower* which provides a smoothed version of the samples for quick reference.
- *Frequency Spectral*:
  AudioSpectrumEnvelope the frequency analysis over the whole signal.
  AudioSpectrumCentroid and *AudioSpectrumSpread* describe the centre of gravity of the frequency spectrum and its spread respectively. The spread helps in distinguishing pure sound from noise.
- *Signal Parameters*:
  AudioFundamentalFrequency which is an estimation of the pitch.
  AudioHarmonicity describing the harmonicity of the signal, which is used to differentiate harmonic sounds (e.g. music) from inharmonic sounds (e.g. bell) from non harmonic sounds (e.g. noise).
- *Silence Segments*: Tagging silent frames of the signal.

Appendix B: Time Plan

B.1. Time Plan



The time plan for project "Speaker Diarization"

Appendix C: Preliminary Results

C.1. Segmentation Results

The results of applying different metrics using DNA representation in a fixed-window metric approach on a TIMIT – based dataset; in addition to the KL2 metric using MFCC representation (see 4.5.); are presented in Figure C.1. Figure C.2 shows the same comparison on a NTIMIT-based dataset. It is clear that the DNA representation is performing better than the MFCC representation under similar conditions. Also, that there is no one best metric for all cases. However, it seems that the modified KL[8] metric is doing well in all cases; therefore, I will use it in the rest of this report.

Figure C.3 and Figure C.4 show similar test results using the anchors model (by selecting 32 hyper-speakers and relating the frames to them) in comparison to the KL metric (Figure C.3 is the results on a TIMIT-based dataset and Figure C.4 is the result on a NTIMIT-based dataset). The results are less accurate (with higher miss rate) because of the loss of information that takes place when reducing the number of features to only 32.

Following a model-based approach I render the results in Figure C.5 and C.6 for TIMIT-based and NTIMIT-based datasets respectively. The model is a Normal distribution that is acquired using a separate set of sentences for each speaker in the stream. Then I use the modified KL metric to measure the distance to each speaker (see 5.5.2)

These results might be improved by introducing a validation step of the hypothesized turn points.

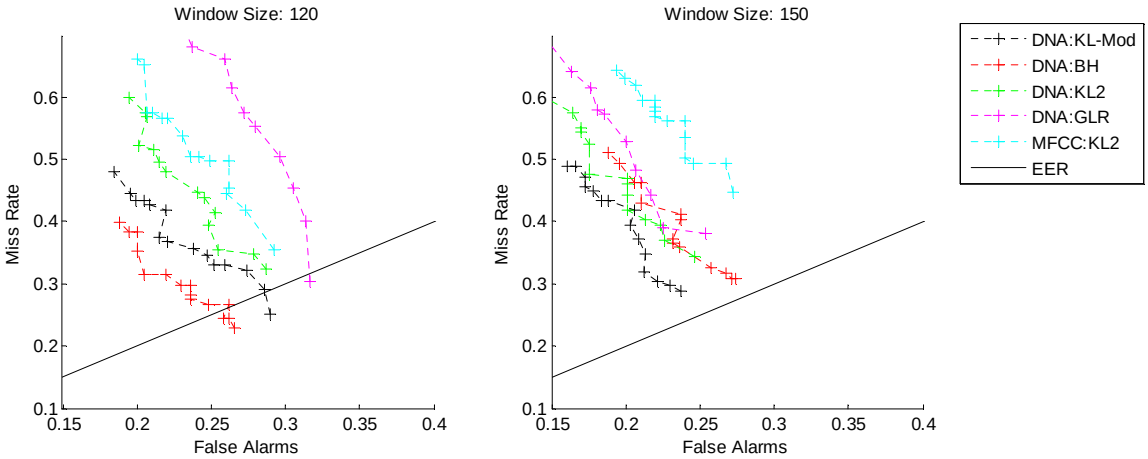


Figure C.1: ROC Curve for metric-based segmentation of a TIMIT-based Dataset

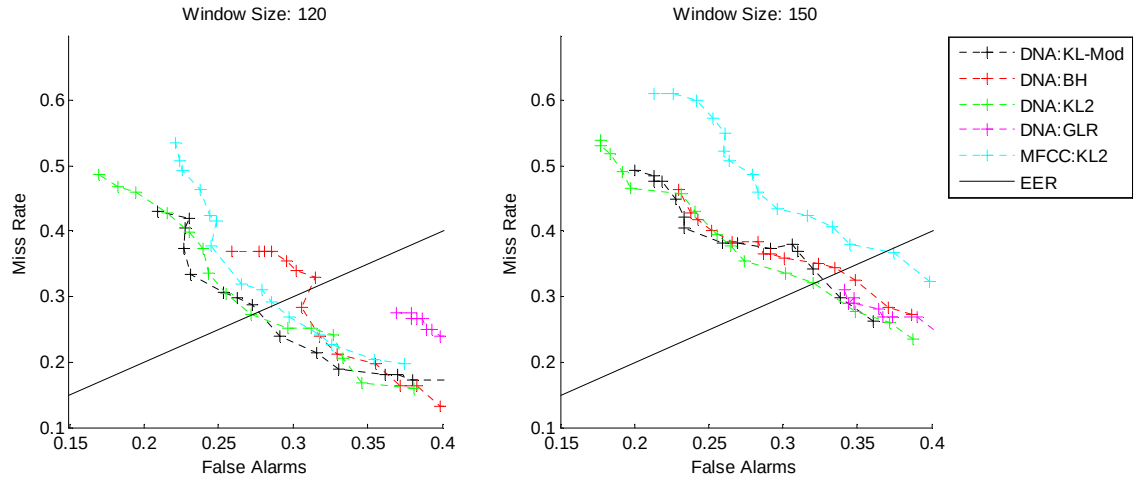


Figure C.2: ROC Curve for metric-based segmentation of a NTIMIT-based Dataset

[8] The modified KL is the same as the KL2 metric after dropping the parts concerning only the covariance.

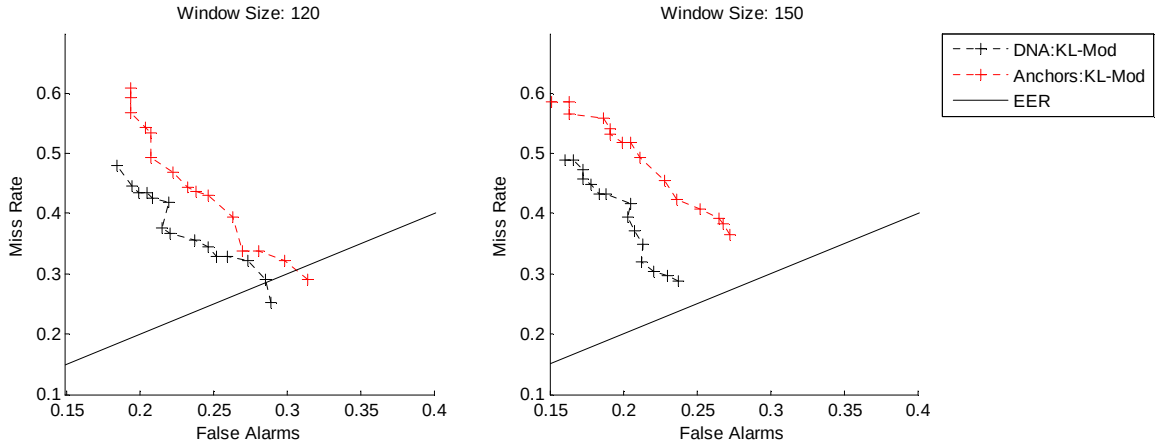


Figure C.3: ROC Curve for metric-based segmentation of a TIMIT-based Dataset using Anchors model

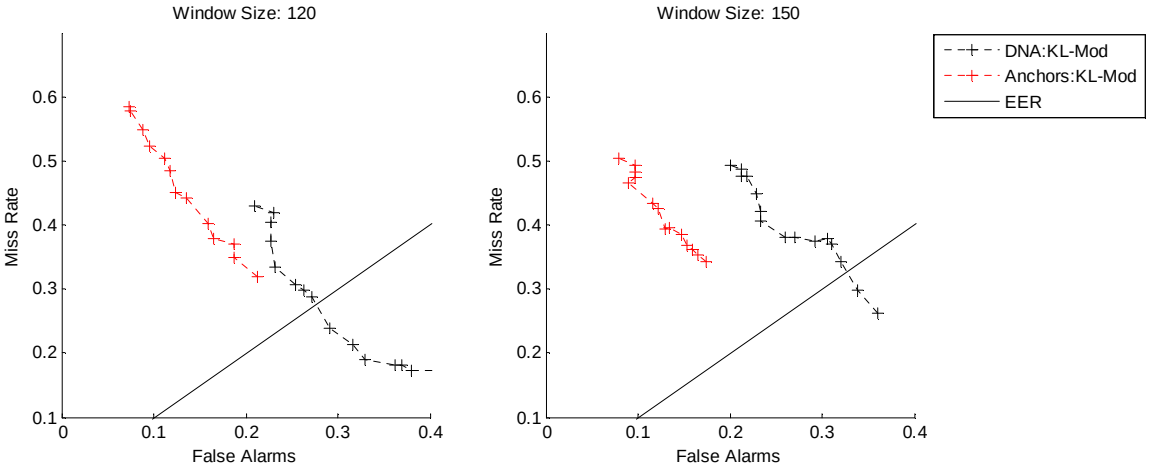


Figure C.4: ROC Curve for metric-based segmentation of a NTIMIT-based Dataset using Anchors model

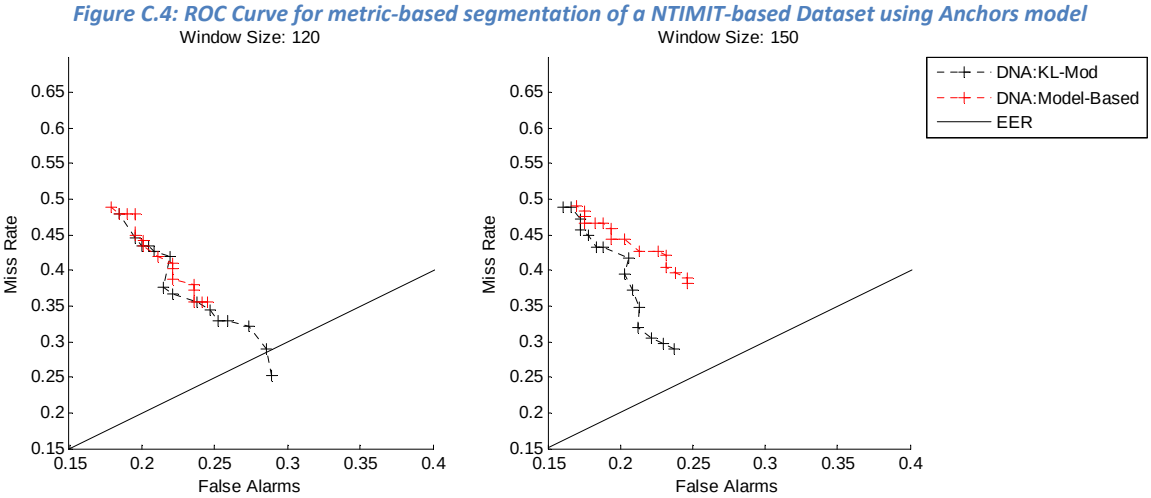


Figure C.5: ROC Curve for model-based segmentation of a TIMIT-based Dataset compared to the metric KL modified

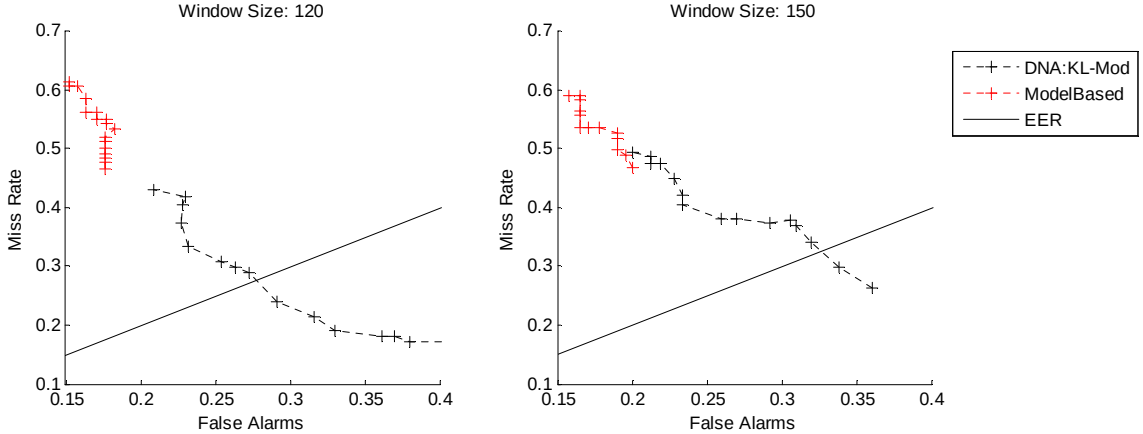


Figure C.6: ROC Curve for model-based segmentation of a NTIMIT-based Dataset compared to the metric KL modified