UNIVERSIDADE TÉCNICA DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Audio Pre-processing and Speech Recognition for Broadcast News

Hugo Daniel dos Santos Meinedo
(Mestre)

Dissertação para obtenção do Grau de Doutor em
Engenharia Electrotécnica e de Computadores

Orientador: Prof. João Paulo da Silva Neto

Júri:

Presidente: Reitor da Universidade Técnica de Lisboa
Vogais: Doutor Jean-Pierre Martens
Doutor Luís Henrique Martins Borges de Almeida
Doutora Isabel Maria Martins Trancoso
Doutor António Joaquim dos Santos Romão Serralheiro
Doutor João Paulo da Silva Neto

Janeiro 2008

Abstract

This thesis presents part of the work done in the development of a fully functional prototype system for the selective dissemination of multimedia information. Our media monitoring prototype was developed for Broadcast News (BN) data, specifically for TV news shows. This thesis had four different main tasks: 1. definition and collection of appropriate BN speech resources, 2. development of Automatic Speech Recognition (ASR) acoustic models appropriate for the BN task, 3. development of Audio Pre-Processing (APP) algorithms for partitioning and classifying the audio stream and 4. development of a media monitoring prototype joining the core technologies developed.

Resumo

Nesta tese é apresentado parte do trabalho de desenvolvimento de um protótipo 100% funcional para a disseminação selectiva de informação multimédia. O nosso sistema de monitorização de media foi desenvolvido para transmissões noticiosas, mais especificamente para programas noticiosos televisivos. Esta tese é composta por quatro grandes tarefas distintas: 1. definição e recolha de recursos de fala de transmissões noticiosas televisivas apropriados, 2. desenvolvimento de modelos acústicos para o reconhecimento de fala contínua apropriados para esta tarefa, 3. desenvolvimento de algoritmos para o pré processamento do áudio que permitem particionar e classificar adequadamente o sinal áudio complexo e 4. implementação de um protótipo para monitorização de dados multimédia que juntou com sucesso as tecnologias desenvolvidas.

Keywords

Audio Pre-processing

Automatic Speech Recognition

Broadcast News

Selective Dissemination of Multimedia information

Bayesian Information Criterion

Multilayer Perceptrons

Palavras chave

Pré-processamento Audio

Reconhecimento automático de fala

Transmissões Noticiosas

Disseminação selectiva de informação multimedia

Critério de Informação Bayesiano

Perceptrões Multicamada

USA congress library catalogue identification:
TK7882.S65 -- 1. Speech processing systems. 2. Automatic Speech Recognition

Acknowledgements

The author was sponsored by a FCT scholarship (SFRH/BD/6125/2001). This work was partially funded by FCT projects POSI/PLP/47175/2002 and POSC/PLP/58697/2004. INESC-ID Lisboa had support from the POSI program of the ”Quadro Comunitário de Apoio III”.

This thesis would not have been possible without the motivation, support and encouragement that I have received during these five long years. I would like to thank and acknowledge to all that have contributed to my work.

Em primeiro lugar quero agradecer ao Professor João Paulo Neto pela sua orientação, apoio, colaboração e motivação, além de ter sido um dos principais responsáveis pelo meu interesse no processamento computacional de fala.

Agradeço também à Professora Isabel Trancoso pela disponibilidade e ajuda sempre pronta além de me ter desvendado os mistérios do processamento de fala na sua disciplina de Mestrado.

Gostaria também de agradecer especialmente ao Professor Luís Borges Almeida pelos ensinamentos, conselhos e ideias sempre pertinentes.

Agradeço também a todos os meus colegas do laboratório de sistemas de linguagem falada (L2F) do INESC-ID Lisboa pela colaboração, ajuda indispensável, camaradagem e excelente ambiente de trabalho proporcionado.

Também um agradecimento aos meus pais pelo apoio incondicional em toda a minha vida e em particular nesta tese.

Finalmente, um agradecimento muito especial é dedicado à minha mulher por toda a paciência e sobretudo por me ter dado motivação para concluir este trabalho.

Thank you all!

Contents

List of Figures

List of Tables

List of Acronyms

Letter	Acronym	Expansion
A		
	AC	Acoustic Model
	ACD	Acoustic Change Detection
	ALERT	European Project (name)
	ANN	Artificial Neural Network
	APP	Audio Pre-Processing
	ASR	Automatic Speech Recognition
	AVI	Video format file
B		
	BIC	Bayesian Information Criterium
	BC	Background Conditions classification
	BLAS	Basic Linear Algebra Sub-programs
	BN	Broadcast News
C		
	CD	Context Dependent
	CER	Classification Error Rate
	CI	Context Independent
D		
	DER	Diarization Error Rate
E		
	EER	Equal Error Rate
	ESTER	French Evaluation campaign (name)
F		
	F0	Baseline broadcast speech (clean, planned speech)
	FSM	Finite State Machine / Models
G		
	gi	Gender Independent
	gd	Gender Dependent
	GD	Gender Detection
	GMM	Gaussian Mixture Models
H		
	HMM	Hidden Markov Model
	HTML	HyperText Markup Language

Table 1: Acronyms used during this PhD report.

Letter	Acronym	Expantion
J		
	JD	Jingle Detection
	JER	Jingle Error Rate
K		
	KL2	Kullback-Liebler (simetric)
L		
	L2F	Spoken Language Systems Laboratory (name)
	LLR	Likelihood Linear Regression
	LM	Language Model
	Log-RASTA	Logaritmic Relative Spectral Transform
M		
	MAP	Maximum A Posteriori
	MDE	Meta Data Extraction
	MFCC	Mel Frequency Cepstral Coefficients
	ML	Maximum Likelihood
	MLLR	Maximum Likelihood Linear Regression
	MLP	Multi Layer Perceptron
	MMI	Maximum Mutual Information
	MP3	Audio format file
	MPEG-x	Compressed video format file
	MSG	Modulation SpectroGram
N		
	NER	News Error Rate
	NIST	National Institute of Standards and Technology (name)
O		
	OOV	Out Of Vocabulary word
P		
	PDA	Personal Digital Assistant (Portable computer)
	PER	Purity Error Rate
	PLP	Perceptual Linear Prediction coefficients
	POS	Part Of Speech (language model)
R		
	RM	Real Media
	RT	Rich Text
	RT	Real Text
	RT-04F	Real Text 2004 Fall Evaluation campaign (name)

Table 2: Acronyms used during this PhD report.

Letter	Acronym	Expantion
S		
	SI	Speaker Independent
	SID	Speaker Identification
	SC	Speaker Clustering
	SMIL	Synchronized Multimedia Integration Language
	SNR	Signal to Noise Ration
	SNS	Speech / Non-speech classification
	SR	Speech Recognition
	STT	Speech To Text
	SU	Sentence like Units
T		
	T2	Hotelling's T-square statistic
	TD	Topic Detection
	TI	Topic Indexation
	TN	Text News
	TS	Topic Segmentation
	TV	Television
X		
	XML	Extended Markup Language
	xRT	times Real Time
W		
	WAV	Windows Audio format file
	WER	Word Error Rate
	WFST	Weighted Finite State Transducer

Table 3: Acronyms used during this PhD report.

Chapter 1

Introduction

This thesis presents part of the work done in the development of a fully functional prototype system for the selective dissemination of multimedia information. The idea originally developed within the European Project ALERT (Alert system for selective dissemination of multimedia information) was to build a system capable of identifying specific information in multimedia data consisting of audio-visual streams. To accomplish this goal our laboratory has been working in the development of the core technologies that constitute a media monitoring system, namely Audio Pre-Processing (APP), Automatic Speech Recognition (ASR) and Topic Detection (TD) algorithms.

The last years show a large demand for the monitoring of multimedia data. A good example of this multimedia data is Broadcast News (BN) streams. Every day thousands of TV and radio stations broadcast many hours of information (news, interviews, documentaries). In order to search and retrieve information from these massive quantities of data it is necessary to have it stored in a database, and most important, to have it indexed and catalogued.

This chapter introduces the core technologies behind our media monitoring system, explains our motivation for the thesis work, enumerates what has been done and gives an overview of the state of the art systems for APP and ASR. Finally the chapter ends with our main contributions and a brief description of how this thesis document is organized.

1.1 Media Monitoring system components

Our media monitoring prototype system for the selective dissemination of multimedia information was developed for BN data, specifically for TV news shows. From a set of news shows to monitor from different BN TV stations the system has registered users with a profile regarding the news topics that are of his/her interest. Then the system indexes and catalogues a news show and compares the generated story topics with the user profiles. If there are matches it sends alert emails to the users with the title, short summary and video link to the relevant news detected (Figure 1.1).

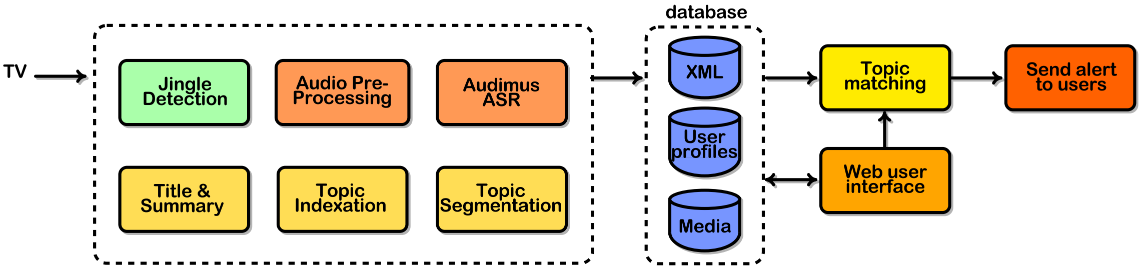

Figure 1.1: Media Monitoring system functional block diagram

Our media monitoring system is composed by several modules whose algorithms index and catalogue BN data. The most important ones are the Audio Pre-Processing (APP), Automatic Speech Recognition (ASR) and Topic Segmentation and Indexation (Topic Detection, TD). This thesis describes the work done in the development of APP and ASR modules.

1.1.1 Automatic Speech Recognition

Automatic Speech Recognition (ASR) can be described as the process of machine transcriptioning into text the words uttered by human speakers. If one wants to perform a query in a database of BN data it is necessary to have a textual transcription or at least a description of the news content. Modern ASR systems use hierarchical models to recognize speech. The input audio is broken into small chunks and the **acoustic model** of the ASR system attributes to each chunk a probability or likelihood of occurrence of a basic speech sound (phoneme). Then a search guided by two models determines the final sequence of words uttered. The two models are the **lexicon** which models the sequence of phonemes that form words and the **language model** which models the sequence of words of the language.

Over the years our laboratory has developed ASR systems for different tasks evolving from simpler to more complex needs: isolated digits and connected word recognition, continuous recognition systems for dictation tasks and more recently continuous speech recognition systems for Broadcast News (BN) tasks. For the recognition of Broadcast News training appropriate ASR models involves large amounts of data both audio for the **acoustic model** and text for the **language model**. Coping with large amounts of audio data is an issue which was addressed during this thesis work more specifically developing efficient training methods for the **acoustic model**.

1.1.2 Audio Pre-Processing

Audio Pre-processing (APP) is the task of partitioning and classifying an audio stream (Figure 1.2). In our system this is divided into the following tasks:

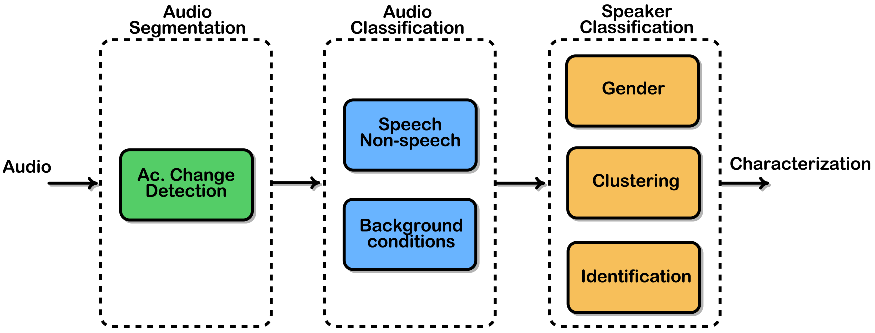

Figure 1.2: Audio Pre-Processing block diagram

- Acoustic Change Detection (ACD)

- Speech/Non-speech classification (SNS)
- Gender Detection (GD)
- Background Conditions classification (BC)
- Speaker Clustering (SC)
- Speaker Identification (SID)

Acoustic Change Detection (ACD) is the task responsible for the detection of audio locations where speakers or background conditions have changed. Speech/Non-speech (SNS) classification is responsible for determining if the audio contains speech or not. Gender Detection (GD) distinguishes between male and female gender speakers. Background Conditions (BC) classification indicates whether the background is clean, has noise or music. Speaker Clustering (SC) identifies all the speech segments produced by the same speaker. Speaker Identification (SID) is the task of detecting the identity of certain often recurring speakers like news anchors or very important personalities.

The problem of distinguishing speech signals from other non-speech signals is becoming increasingly important as ASR systems are being applied to an increasing number of real-world multimedia applications. Furthermore, audio and speech segmentation will always be needed to break the continuous audio stream into manageable chunks. By using ASR acoustic models trained for a particular acoustic condition such as male speaker versus female speaker or wide bandwidth (high quality microphone input) versus telephone narrow bandwidth the overall performance can be significantly improved. Finally this pre-processing could also be designed to provide additional interesting information such as division into speaker turns and speaker identities allowing for automatic indexing and retrieval of all occurrences of the same speaker. It can also provide end of turn information relevant for recovering punctuation marks, syntactic parsing, etc. Additionally speaker segmentation and clustering information can be used for efficient speaker adaptation which has been shown to significantly improve ASR accuracy [Chen and Gopalakrishnan, 1998, Johnson and Woodland, 1998, Ramabhadran et al., 2003]. Finally, all this information, when combined with the text output of the ASR, results in a rich transcription which is much easier to understand.

1.1.3 Topic Detection

Topic Detection (TD) is the task of dividing the BN stream into different stories and classifying those stories according to a set of news topics. In our media monitoring prototype, TD algorithms were developed by my colegue Rui Amaral [Amaral et al., 2001, Amaral and Trancoso, 2003b, Amaral and Trancoso, 2003c, Amaral and Trancoso, 2003a, Trancoso et al., 2003, Trancoso et al., 2004, Amaral et al., 2006, Amaral et al., 2007]. The TD module uses information from the APP and ASR modules. The TD module provides the top level content classification based on all information collected and is a key step in a content indexing and retrieval system such as our media monitoring prototype. RTP, the Portuguese public broadcast company and our user partner in the ALERT project, was interested in indexing every news story and not only the stories according to certain profiles. To accomplish this indexing task, we based our topic concept in a thematic thesaurus definition that was used at RTP in their manual daily indexing process. This thesaurus follows rules which are generally adopted within EBU (European Broadcast Union) and has an hierarchical structure with 22 thematic areas in the first level. Although each thematic area is subdivided into (up to) 9 lower levels, we implemented only 3 in our system. In fact, it is difficult to represent the knowledge associated with a deeper level of representation due to the relative small training data in our automatic topic indexation method. This structure, complemented with geographic (places) and onomastic (names of persons and organizations) descriptors, makes our topic definition [Trancoso et al., 2003]. The use of this hierarchically organized structure makes our TD system significantly different from the type of work involved in the TREC SDR Track [Garofolo et al., 2000].

Our laboratory TD module can be subdivided into two components, Topic Segmentation (TS) which splits the Broadcast News show into constituent stories and Topic Indexation (TI) which assigns one or multiple topics to each story, according to the thematic thesaurus.

1.2 Motivation

In the Master thesis the author had worked with continuous speech recognition systems for dictation tasks and had developed better acoustic models by using together phonemes and information from larger time spans than phonemes (syllables). For this thesis the original idea was to further pursue the development of new ASR acoustic models using syllabic units instead of phonemes thus developing *context dependent* acoustic models. These new acoustic models would be developed for new and more demanding tasks, namely for BN thus evolving the speech recognition systems from dictation. The first steps of our work were towards the collection of appropriate BN resources that would enable us to develop new ASR systems. For this purpose it was necessary to collect BN corpora, fundamental resources for the development of all modules of a media monitoring system. The thesis work started by the collection of a new European Portuguese BN corpus with the appropriate size for training our BN ASR system and also for training TD algorithms.

In parallel with the collection of the BN database, new acoustic models were developed for the task. Each time more BN training data was available, better and more robust ASR models were created. We also investigated techniques for expanding the ASR acoustic models in order to cope with the increased training data size. It enabled us to have a better acoustic model without increasing too much the total number of parameters of the model. Another important step was to use automatic annotated training data using confidence measures. This enabled us to build gender dependent models and speaker adapted models (for news anchors).

BN audio is composed by very diverse acoustic conditions: speech can be read but also most often is spontaneously spoken with disfluencies, stuttering, hesitations, restarts, chopped words, etc. Background conditions are very important too. The speakers can be inside a studio with very clean background (news anchors) or in the street interviewing someone with lots of background noises. There can be music in the background. There can be long portions of audio without useful speech (music from jingles or commercial breaks). Last, all these difficult conditions can be present at the same time making BN speech recognition a very demanding task. In order to simplify the audio processing, it is necessary to have a pre-processing system that segments and classifies the input audio before it is passed to the speech recognition system.

This thesis thus aims at developing and investigating different APP techniques to segment and classify complex audio signals such as the ones typically found in Broadcast News and comprised of a sequence or a mix of complex signals (such as music, speech from different speakers, different acoustic environments etc.), with a view to further processing, recognition and topic detection. In summary, the following problems involved in the APP module will be one of the main focus of this thesis:

1. Acoustic Change Detection: finding the speaker boundaries i.e. begin and end point for every speaker and finding locations where acoustic conditions change.

2. Speech/Non-speech classification: classify the segments according to the presence of speech.

3. Gender Detection: for the segments classified as containing speech, detect if the speaker is of male or female gender.

4. Background conditions classification: indicate if the background is clean or has noise or music.

5. Speaker clustering: tag with a unique label all speech segments uttered by the same speaker. This also involves trying to find the actual number of speakers in the news

show.

6. Speaker Identification: find the speaker identity from a set of pre-defined speakers.

The development and implementation of this Audio Pre-Processing system (APP) was responsible for the change of direction in this thesis work. Another very important part of this thesis work was the development and implementation of a fully functioning system for the selective dissemination of multimedia information. Our media monitoring prototype joins successfully the core technologies developed during the thesis work and also Topic Detection and Summarization components. In summary, this thesis had four different tasks: 1. definition and collection of a BN database, 2. development of acoustic models appropriate for the BN task, 3. development of an APP system, 4. development of a media monitoring prototype. The research and innovation component is linked with tasks 2. and 3. The development component is related with tasks 1. and 4.

1.3 State of the art

This section discusses the background work related to different Audio Pre-Processing and Automatic Speech Recognition systems. This includes discussion of the algorithms which are most commonly used and the ones that currently achieve the best results (state of the art). This is important to understand where we stand in comparison with other more developed languages in terms of speech technologies like the English language. We start with ASR (Automatic Speech Recognition) state of the art and then proceed to Audio Pre-Processing (APP) state of the art giving an overview for each specific component.

1.3.1 Automatic Speech Recognition

Current state of the art BN ASR systems for the English language have performances for Word Error Rate (WER) less than 13% with 10 xRT [Nguyen et al., 2005] and less than

16% with Real-Time performance [Matsoukas et al., 2005]. These values were obtained in recent NIST RT evaluations as summarized in Figure 1.3.

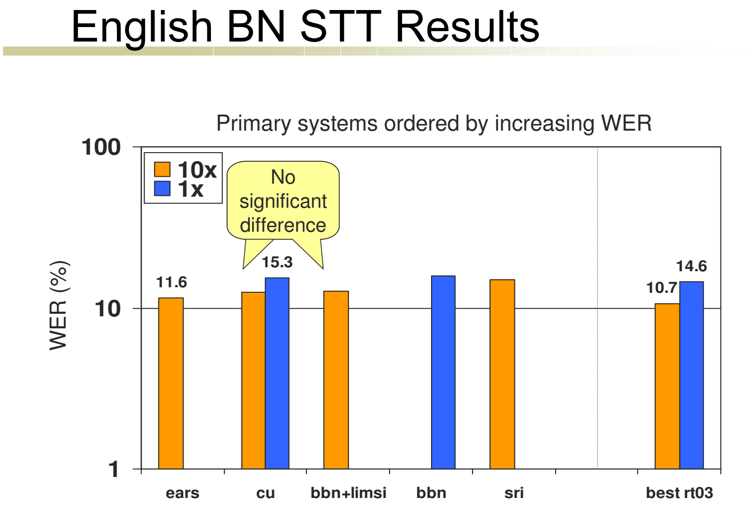

Figure 1.3: NIST BN ASR evaluation for English language (RT-04F)

Table 1.1 represents the ASR system improvement developed at BBN [Nguyen et al., 2005]. As we can see a significant gain was obtained by using a number of small improvements. The bigger ones came from training the acoustic models using a total of 1700 hours of speech.

ASR detail of improvement	WER
Baseline (RT03 trained on 200 h)	13.4
843 h acoustic training	12.1
1700 h acoustic training	11.3
“ + MMI for all models	11.0
“ + duration modelling	10.9
“ + online speaker clustering	10.8
“ + longer utterances (7 s)	10.5
“ + new lexicon and LM	10.4

Table 1.1: BBN ASR system improvements in the RT-04F Dev04 test set.

If we look at the ASR systems for the French language, which is much more similar to the European Portuguese than the English, the state of the art ASR results obtained for the ESTER phase II campaign [Galliano et al., 2005] obtained 11.9% of WER overall. They also obtained around 10% of WER for clean speech (studio or telephone), to be compared with 17.9% in the presence of background music or noise. But this also means that ESTER test data has much more easy (clean) conditions than more difficult ones (more noise). The overview article [Galliano et al., 2005] outlined that in a more detailed analysis of the results, unsurprisingly, systems are sensitive to degraded speech quality and to background noise. The best system which obtained the 11.9% WER uses acoustic models with GMMs trained for estimating triphones. The feature used are 12th order MFCC [Mermelstein, 1976] and the log energy with first and second order derivatives giving a total of 39 coefficients. This feature vector is linearly transformed to better fit the diagonal covariance Gaussians used for acoustic modelling. The acoustic models were trained on about 190 hours of BN training data. For the final decoding pass, the acoustic models include 23k position-dependent triphones with 12k tied states, obtained using a divisive decision tree based clustering algorithm with the 35 phones. Two sets of gender dependent acoustic models were built for each data type (wideband and telephone). Decoding is performed in three passes, where each pass generates a word lattice which is expanded with a 4-gram LM. Then the posterior probabilities of the lattice edges are estimated using the forward-backward algorithm and the 4-gram lattice is converted to a confusion network with posterior probabilities by iteratively merging lattice vertices and splitting lattice edges until a linear graph is obtained. The words with the highest posterior in each confusion set are hypothesized. For the first and second passes the lattice rescoring step is done using a standard 4-gram language model, while in the third pass rescoring is done with the a neural network model and the POS language model. Also to speedup the third pass the search space for each audio segment to be decoded is restricted to the a word graph derived from the lattice generated in the second pass. This results in a decoding speed of about 7.5 xRT. Unsupervised acoustic model adaptation is carried out for each speaker between the decoding passes making use of the hypotheses of the previous pass. This done by means of a constrained MLLR adaptation followed by a un-constrained MLLR. For the regular MLLR adaptation, two regression classes (speech and non-speech) are used in the second pass, whereas a data driven clustering with a variable number of classes is used in the third pass. A real time version of the decoding procedure has also been implemented. For this condition the decoding is reduced to two passes with very tight pruning thresholds (especially for the first pass) and with fast Gaussian computation based on Gaussian short lists. This real time version obtained 16.8% WER in the same ESTER test set.

1.3.2 Acoustic Change Detection

The aim of this step is to find points in the audio stream likely to be change points between audio sources. If the input to this stage is the unsegmented audio stream, then the ACD looks for both speaker and Speech/Non-speech change points. If a SNS or GD classifier has been run first then the ACD looks for speaker change points within each speech segment. Various approaches have been used for ACD in the literature, which can be categorized as follows:

- Energy based segmentation: audio signal energy level is used to distinguish between portions of audio (speech, music, noise) and silence.

- Decoder-guided segmentation: The input stream can first be decoded; then the desired segments can be produced by cutting the input at the silence locations generated from the decoder [Woodland et al., 1997]. Other information from the decoder, such as gender information could also be utilized in the segmentation.

- Model-based segmentation: This involves making different models e.g. GMMs, for a fixed set of acoustic classes, such as telephone speech, pure music, etc. from a training corpus [Bakis et al., 1998, Kemp et al., 2000], the incoming audio stream can be classified by Maximum Likelihood selection over a sliding window; segmentation can be made at the locations where there is a change in the acoustic class.

- Metric-based segmentation: A distance-like metric is calculated between two neighbouring windows placed at each sample; metrics such as Kullback-Liebler (KL2) divergence [Siegler et al., 1997], LLR [Delacourt et al., 2000] or BIC [Chen and Gopalakrishnan, 1998, Vandecatseye and Martens, 2003] can be used. The local maxima or minima of these metrics are considered to be the change points.

The energy based and the decoder guided segmentation only places boundaries at silence locations, which in general has connection with the acoustic changes in the data. The model based segmentation approaches might have problems generalizing to unseen data conditions if the models are no longer compatible with the new data conditions. The metric based approaches generally require threshold/penalty terms to make decisions although they are simpler to build. [Chen and Gopalakrishnan, 1998] formulated the problem of ACD as a model selection problem and used BIC for this purpose. However, a tuneable parameter was introduced in the penalty term. In practice, this parameter is used to improve the performance of the system for a particular condition. This parameter therefore implicitly defines a threshold. This was observed in our own experiments and also in several other works [Tritschler and Gopinath, 1999, Delacourt and Wellekens, 2000, Vandecatseye and Martens, 2003]. Later, [Tritschler and Gopinath, 1999] proposed several heuristics not only to make the algorithm of [Chen and Gopalakrishnan, 1998] faster, but also to give importance to detecting short changes (less than 2 seconds). There is also works which attempts to make the detection procedure faster by applying a distance measure prior to BIC. DISTBIC was another work in this direction by [Delacourt and Wellekens, 2000]. An LLR based distance computation prior to BIC was proposed in this work, which is faster than BIC. Then, only selected change points were passed through the BIC test. In another work in this direction, [Zhou and Hansen, 2000] proposed applying T 2-statistics prior to BIC. [Vandecatseye and Martens, 2003] proposed a modification to BIC in terms of normalizing the BIC score and showed it to be better than the non-normalized BIC. However, researchers have also explored techniques other than BIC and LLR for this task. [Gish et al., 1991] proposed a distance measure for this purpose, which is referred to as Gish-distance in the literature. This distance is also based on LLR and was also used by [Kemp et al., 2000]. In an interesting study, [Kemp et al., 2000] observed that model based approaches achieve high precision at moderate recall while metric based approaches are capable of achieving high recall at moderate precision. Thus they combined the two approaches. [Liu and Kubala, 1999] introduced a new penalized LLR (BIC-like) criterion for speaker change detection. Another interesting point in [Liu and Kubala, 1999] was that the change points were detected based on the output of a speech recognizer. However, this approach also used thresholds, which were set to minimize the total error (false acceptance and false rejection).

State of the art systems in recent evaluation campaigns use simpler metrics based approaches [Sinha and et al., 2005, Zhu and et al., 2005] to over-segment the speech data. Later the segmentation is refined using Speaker Clustering.

Alternatively, or in addition, a word or phone decoding step with heuristic rules may be used to help find speaker change points such as in [Liu and Kubala, 1999, Tranter and Reynolds, 2004]. However, this approach can over-segment the speech data and requires some additional merging or clustering to form viable speech segments, and can miss boundaries in fast speaker interchanges if relying on the presence of silence or gender changes between speakers.

Best published results for this task are around 75% F-measure (explained in Chapter 4). But normally the researchers do not include ACD results in their papers and prefer to give system overall results in terms of Diarization Error Rate, measuring the system performance to SC.

1.3.3 Speech/Non-speech classification

The aim of this step is to find the regions of speech in the audio stream. Non-speech regions to be discarded can consist of many acoustic phenomena such as silence, music, room noise, background noise, or cross-talk. Earlier work regarding the separation of speech and non-speech (noise, music) classes addressed the problem of classifying known homogenous segments as speech or noise/music. Earlier research also focused more on devising and evaluating characteristic features for classification. [Sheirer and Slaney, 1997] investigated the use of ”low-energy” frames, spectral roll-off points, spectral centroid (correlate of zero crossing rate), spectral flux (delta spectral magnitude) and 4 Hz modulation energy (syllabic rate of speech). [El-Maleh et al., 2000] used line spectral frequencies for SNS discrimination. More recently, in a completely different approach, [Williams and Ellis, 1999] used posterior probability based features for this purpose and [Zibert et al., 2006] used phoneme recognition based features. Generally in more recent works, cepstral coefficients like PLP [Hermansky et al., 1992] and MFCC [Mermelstein, 1976] are used for classifying speech and non-speech signals rather than using specially devised features such as those mentioned for early systems. In fact these Cepstral coefficient features are also widely used in the other types of audio pre-processing classification like gender, background and speaker clustering. Nevertheless they are smoothed to hide or remove aspects of the signal that are not phonetically relevant, such as speaker identity and background noise so they might not be the best basis for distinguishing between different speakers or backgrounds. The general approach used in SNS systems is maximum-likelihood classification with Gaussian mixture models (GMMs) trained on labelled training data to distinguish between several classes of audio [Vandecatseye and Martens, 2003, Sinha and et al., 2005, Zhu and et al., 2005], although different class models can be used, such as multistate HMMs. The simplest system uses just SNS models while others use four speech models are used for the possible gender/bandwidth combinations. Noise and music are explicitly modelled in [Gauvain et al., 1998, Zhu and et al., 2005] which have classes for speech, music, noise, speech + music, and speech + noise, while [Sinha and et al., 2005] uses wideband music + speech, narrowband speech, music and speech. The extra speech + other models are used to help minimize the false rejection of speech occurring in the presence of music or noise, and this data is subsequently reclassified as speech. The classes can also be broken down further, as in [Liu and Kubala, 1999], which has eight models in total, five for non-speech (music, laughter, breath, lip-smack, and silence) and three for speech (vowels and nasals, fricatives, and obstruents). When operating on unsegmented audio, Viterbi segmentation, (single pass or iterative with optional adaptation) using the models is employed to identify speech regions. If an initial segmentation is already available (from ACD), each segment is individually classified. Silence can be removed in this early stage using a phone recognizer (as in [Sinha and et al., 2005]). For BN audio, SNS detection performance is around 3% CER, typically less than 1% miss (speech in reference but not in the hypothesis) and 1% 2% false alarm (speech in the hypothesis but not in the reference). When the SNS detection phase is run early in a system, or the output is required for further processing such as for transcription, it is more important to minimize speech miss than false alarm rates, since the former are unrecoverable errors in most systems. However, the DER, used to evaluate SC performance, treats both forms of error equally.

1.3.4 Gender detection

The aim of this stage is to partition the segments into common groupings of gender (male or female). This is done to reduce the load on subsequent clustering, provide more flexibility in clustering settings (for example female speakers may have different optimal parameter settings to male speakers), and supply more side information about the speakers in the final output. If the partitioning can be done very accurately and assuming no speaker appears in the same broadcast in different classes (for example both in the studio and via a pre-recorded field report) then performing this partitioning early on in the system can also help improve performance while reducing the computational load. The potential drawback in this partitioning stage, however, is if a subset of a speaker's segments is misclassified the errors can be unrecoverable, although it is possible to allow these classifications to change in a subsequent re-segmentation stage. Classification for gender is typically done using maximum-likelihood classification with GMMs (Gaussian Mixture Models) [Zdansky et al., 2004] or Artificial Neural Network (ANN) [Meinedo and Neto, 2005] trained on labelled training data. This can be done either in conjunction with the SNS detection process or after the initial ACD segmentation. An alternative method of gender classification, used in [Sinha and et al., 2005], aligns the word recognition output of a fast ASR system with gender dependent models and assigns the most likely gender to each segment. This has a high accuracy but is unnecessarily computationally expensive if a speech recognition output is not already available and segments ideally should be of a reasonable size (typically between 1 and 30 s). Gender Classification Error Rates (CER) are around 5% for BN audio with best published results around 2%.

1.3.5 Background Conditions classification

Background Conditions classification is very similar to GD in terms of philosophy except it normally has more distinct classes for instance clean, music and noise than gender which normally has only male and female. State of the art solutions to this problem use model based approaches with GMM [Zibert et al., 2005]. Unlike GD there are very few examples of BC modules in literature. Recently with the ESTER French BN campaign [Galliano et al., 2005], which had a task for music and speech tracking (read bellow) some French institutions developed modules for detecting the presence of music in the foreground and background [Istrate et al., 2005]. Results are normally inferior since this is a difficult task.

1.3.6 Speaker Clustering

The purpose of this stage is to associate or cluster segments from the same speaker together. The clustering ideally produces one cluster for each speaker in the audio with all segments from a given speaker in a single cluster. The predominant approach used by APP state of the art systems for this problem is hierarchical agglomerative clustering approach i.e. starting from a large number of segments (clusters), some of the clusters are merged following an appropriate strategy. This strategy mostly consists of computing a distance metric between two clusters and then merging the clusters with the minimum distance. Since most of the popular distance matrices are monotonic functions of the number of clusters, an external method of controlling the number of clusters (or merging process) is also necessary and part of the problem. Thus, most of the previous work on this topic revolves around employing a suitable distance metric and corresponding stopping criterion. The hierarchical agglomerative clustering consists of the following steps: 1) initialize leaf clusters of tree with speech segments; 2) compute pair-wise distances between each cluster; 3) merge closest clusters; 4) update distances of remaining clusters to new cluster; 5) iterate steps 2) 4) until stopping criterion is met. The clusters are generally represented by a single full covariance Gaussian [Moh et al., 2003, Sinha and et al., 2005, Zhu and et al., 2005, Liu and Kubala, 2003], but GMMs have also been used [Moraru et al., 2004], sometimes being built using mean-only MAP adaptation of a GMM of the entire test file to each cluster for increased robustness.

One of the earliest pieces of work on SC from the point of view of speaker adaptation in ASR systems was proposed by [Jin et al., 1997]. In this work, the Gish-distance proposed in [Gish et al., 1991] was used as a distance matrix, which is based on Gaussian models of the acoustic segments. Hierarchical clustering was performed on this distance matrix while selecting the best clustering solution automatically by minimizing the within-cluster dispersion with some penalty against too many clusters. In [Siegler et al., 1997] used KL2 divergence (relative cross entropy) as the distance metric for this purpose. The KL2 distance between the distributions of two random variables A and B is an information theoretic measure equal to the additional bit rate needed when encoding random variable B with a code that was designed for optimal encoding of A. In an agglomerative clustering approach, the KL2 distance was also compared with the Mahalanobis [Mahalanobis, 1936] distance. In this framework, an utterance was clustered with an existing cluster if it was within a threshold distance, otherwise it was used as the seed of a new cluster. A top-down split-and-merge clustering framework was proposed in [Johnson and Woodland, 1998]. This work was based upon the idea that the output of this clustering was to be used for Maximum Likelihood Linear Regression (MLLR) adaptation, and so a natural evaluation metric for clustering is the increase in the data likelihood from adaptation. This clustering algorithm was applied to the HTK broadcast news transcription system [Woodland et al., 1997]. The most commonly used distance metric for this purpose is BIC. BIC was proposed for speaker clustering by [Chen and Gopalakrishnan, 1998] who formulated the problem of speaker clustering as model selection. In this work, starting from each segment (hand segmented) as a cluster, hierarchical clustering was performed by calculating the BIC measure for every pair of clusters and merging the two clusters with the highest BIC measure. The clustering was stopped when no two clusters resulted into an increase in BIC measure, when merged. Although, in this work the authors used a theoretically motivated penalty value, subsequent work on speaker clustering using BIC [Tritschler and Gopinath, 1999, Vandecatseye and Martens, 2003] found that adjusting this penalty not only produces better results, but is also necessary for the system to run on unseen data. Adding a Viterbi re-segmentation between multiple iterations of clustering [Zhu and et al., 2005] has also been used to increase performance at the penalty of increased computational cost.

Regardless of the clustering employed, the stopping criterion is critical to good performance and depends on how the output is to be used. Under-clustering fragments speaker data over several clusters, while over-clustering produces contaminated clusters containing speech from several speakers. For indexing information by speaker, both are suboptimal. However, when using cluster output to assist in speaker adaptation of speech recognition models, under-clustering may be suitable when a speaker occurs in multiple acoustic environments and over-clustering may be advantageous in aggregating speech from similar speakers or acoustic environments.

For some applications, it can be important to produce speaker labels immediately without collecting all of the potential data from a particular scenario, for example real-time captioning of a broadcast news show. This constraint prevents the standard hierarchical clustering techniques being used, and instead requires the clustering to be performed sequentially or online. An elegant solution to this, described in [Liu and Kubala, 2003], takes the segments in turn and decides if they match any of the existing speaker clusters using thresholds on distance metrics based on the generalized likelihood ratio and a penalized within-cluster dispersion. If a match is found, the statistics of the matched cluster are updated using the new segment information, whereas if no match is found, the segment starts a new speaker cluster. This process is much faster than the conventional hierarchical approach, particularly when there are a large number of initial segments, and has been used for both finding speaker turns [Liu and Kubala, 2003] and for speaker adaptation within a real-time speech recognition framework [Liu and Kubala, 2005].

1.3.7 Speaker Identification

Although current APP systems are only evaluated using “relative” speaker cluster labels (such as ”speaker23”), it is often possible to find the true identities of the speakers such as “José Alberto Carvalho” (news anchor) or “Cavaco Silva” (Portuguese President). This can be achieved by a variety of methods, such as building speaker models for people who are likely to be in the news show (such as prominent politicians or main news anchors and reporters) and including these models in the speaker clustering stage or running speaker-tracking systems. An alternative approach, introduced in [Lamel et al., 2004] uses linguistic information contained within the transcriptions to predict the previous, current, or next speaker. An extension of this system described in [Tranter, 2004], learns many rules and their associated probability of being correct automatically from the training data and then applies these simultaneously on the test data using probabilistic combination. Using automatic transcriptions and automatically found speaker turns naturally degrades performance but potentially 85% of the time can be correctly assigned to the true speaker identity using this method. Although primarily used for identifying the speaker names given a set of speaker clusters, this technique can associate the same name for more than one input cluster and, therefore, could be thought of as a high-level cluster-combination stage. In [Moraru et al., 2005] it is reported that running SID before SC can obtain important improvements in DER.

1.3.8 Evaluation campaigns

Recently, in the framework of the DARPA-EARS program, NIST started the Rich Text (RT) evaluation [NIST RT, 2003] campaign. The motivation is to produce richer more human readable and easily understandable ASR transcriptions. There are two evaluations, one for STT (speech to text) the same as ASR and another evaluation for Meta Data Extraction (MDE). The MDE evaluation had two subgroups, one concerning structural metadata, that is, sentence units boundary detection and speech disfluencies detection (edit word, filler word and interruption-points). The other subgroup concerns audio diarization metadata, that is, Audio Pre-Processing (commonly known as “who said what?”). The most recent evaluation campaign for Broadcast News data has been conducted in the fall of 2004 and was called NIST RT-04F (Rich Text 04 Fall). The evaluation test data is comprised of 12 audio files each lasting approximately 30 min recorded from news shows of 6 different TV stations. For this test set best evaluation results obtained around 9% DER.

Technolangue ESTER [Galliano et al., 2005] is a similar effort to the NIST RT campaigns but for French BN data. Phase II was held in January 2005. There are two evaluations, one for transcription and the other for segmentation. The later had 3 tasks: sound event tracking, speaker diarization and speaker tracking.

The sound event tracking task consists on identifying, on the one hand, parts of the document containing music, whether in the foreground or in the background, and, on the other hand, parts of the document containing speech, possibly with background music. The results are good for speech detection where very low miss detection rates are achieved (around 2%). This is due to the fact that most systems where tuned to detect speech accurately as a front-end of the transcription system. The music detection task is particularly difficult (around 50% error rates) when the SNR of the music is low.

Speaker tracking is somewhat similar to sound event tracking with speakers being the events to track. The task consists in detecting portions of the document that have been uttered by a given speaker known beforehand and for which training data are available before the test stage.

Speaker diarization aims at segmenting documents into speaker turns and at grouping together portions of the document uttered by the same speaker. Speakers are not known beforehand and identification is not required. Systems must return a segmentation of the document with a possible arbitrary speaker identifier for each segment. As in the speaker tracking task case, a detailed analysis of the results outline results highly dependent on the show, with error rates ranging from 1.5% to 26.1% for a particular system. So, despite the good level of performance reached (11.5%), the systems are very dependent to the nature of the show. This point is clearly linked to the variable nature of the shows (from 14 minutes duration with 5 speakers to 1 hour duration and 60 speakers).

1.4 Contributions

This section looks at various contributions of the author which are described in this document. The definition and collection of very important speech and text resources which drove the development of our audio pre-processing, automatic speech recognition, topic detection and media monitoring systems for BN. The ALERT BN speech corpora for training APP, ASR and TD systems, the COST278-BN database for the APP modules evaluation, the SSNT-BN daily BN news show database and the WEBNEWS-PT daily web Newspaper Text database.

Other very important contribution was the participation in building a fully working prototype for the selective dissemination of multimedia information. The author developed the following modules in this prototype (described in detail throughout this document): the whole CAPTURE block and in the PROCESSING block the jingle Detection module, Audio Pre-Processing module and acoustic models for our BN ASR system. This included better acoustic models for BN task, unsupervised annotation of more training data, gender dependent models and speaker dependent models (for news anchors).

1.4.1 Published articles

During this work the author has published a number of articles describing the work done in Audio Pre-Processing algorithms, in acoustic modelling for ASR and in building our media monitoring prototype. A complete list of articles is given below all published in international conferences and one in a journal.

- Meinedo, H., Souto, N., and Neto, J. (2001). *Speech Recognition of Broadcast News for the European Portuguese Language*. In Proceedings ASRU Workshop 2001, Madonna di Campiglio, Trento, Italy.
- Amaral, R., Langlois, T., Meinedo, H., Neto, J., Souto, N., and Trancoso, I. (2001). *The development of a Portuguese version of a Media Watch system*. In Proceedings EUROSPEECH 2001, Aalborg, Denmark.
- Souto, N., Meinedo, H., and Neto, J. (2002). *Building Language Models for Continuous Speech Recognition Systems*. In Proceedings Portugal for Natural Language

Processing – PorTAL 2002, Faro, Portugal.

- Serralheiro, A., Caseiro, D., Meinedo, H. and Trancoso, I. (2002). *Word Alignment in Digital Talking Books Using WFSTs*. In Proceedings ECDL 2002 – 6th European Conference on Digital Libraries, Rome, Italy.

- Trancoso, I., Neto, J., Meinedo, H., and Amaral, R. (2003). *Evaluation of an Alert System for Selective Dissemination of Broadcast News*. In Proceedings EUROSPEECH 2003, Geneve, Switzerland.

- Neto, J., Meinedo, H., Amaral, R., and Trancoso, I. (2003). *The development of an Automatic System for Selective Dissemination of Multimedia Information*. In Proceedings Third International Workshop on Content-Based Multimedia Indexing – CBMI 2003, Rennes, France.

- Meinedo, H. and Neto, J. (2003). *Audio Segmentation, Classification and Clustering in a Broadcast News task*. In Proceedings ICASSP 2003, Hong Kong, China.

- Neto, J., Meinedo, H., Amaral, R., and Trancoso, I. (2003). *A System for Selective Dissemination of Multimedia Information*. In Proceedings ISCA Workshop on Multilingual Spoken Document Retrieval – MSDR 2003, Hong Kong, China.

- Meinedo, H. and Neto, J. (2003). *Automatic Speech Annotation and Transcription in a Broadcast News task*. In Proceedings ISCA Workshop on Multilingual Spoken Document Retrieval – MSDR 2003, Hong Kong, China.

- Meinedo, H., Caseiro, D., Neto, J., and Trancoso, I. (2003). *AUDIMUS.MEDIA: A Broadcast News Speech Recognition System for the European Portuguese Language*. In Proceedings 6th International Workshop on Computational Processing of the Portuguese Language – PROPOR 2003, Faro, Portugal.

- Meinedo, H. and Neto, J. (2004). *Detection of Acoustic Patterns in Broadcast News using Neural Networks*. In Proceedings Acoustics European Symposium – Acustica 2004, Guimarães, Portugal.

- Trancoso, I., Neto, J., Meinedo, H., Amaral, R., and Caseiro, D. (2004). *An Acoustic driven Media Watch System*. In Proceedings Acoustics European Symposium – Acustica 2004, Guimarães, Portugal.

- Vandecatseye, A., Martens, J., Neto, J., Meinedo, H., Mateo, C., Dieguez, J., Mihelic, F., Zibert, J., Nouza, J., David, P., Pleva, M., Cizmar, A., Papageorgiou, H., and Alexandris, C. (2004). *The COST278 pan-european broadcast news database*. In Proceedings LREC 2004, Lisbon, Portugal.

- Meinedo, H. and Neto, J. (2005). *A stream-based Audio Segmentation, Classification and Clustering Pre-processing System for Broadcast News using ANN models*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

- Zibert, J., Mihelic, F., Martens, J., Meinedo, H., Neto, J., Docio, L., Garcia-Mateo, C., David, P., Nouza, J., Pleva, M., Cizmar, A., Zgank, A., Kacic, Z., Teleki, C., and Vicsi, K. (2005). *The COST278 Broadcast News Segmentation and Speaker Clustering evaluation - overview, methodology, systems, results*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

- Amaral, R., Meinedo, H., Caseiro, D., Trancoso, I., and Neto, J. (2006). *Automatic vs. Manual Topic Segmentation and Indexation in Broadcast News*. In Proceedings IV Jornadas en Tecnologia del Habla, Saragoza, Spain.

- Amaral, R., Meinedo, H., Caseiro, D., Trancoso, I. and Neto, J. (2007). *A Prototype System for Selective Dissemination of Broadcast News in European Portuguese*. In EURASIP Journal on Advances in Signal Processing, article ID 37507.

- Neto J., Meinedo, H., Viveiros, M., Cassaca, R., Martins, C. and Caseiro, D. (2008). *Broadcast News Subtitling system in Portuguese*. To appear in Proceedings ICASSP 2008, Las Vegas, USA.

1.5 Organization

This document is organized as follows:

- Chapter 2 covers the work done in the definition and collection of our BN corpora, fundamental resources for training and evaluating all the algorithms and systems developed during this thesis.
- Chapter 3 describes the development and adaptation to a BN task undertaken by our ASR system. A detailed description of the adaptation steps necessary for our ASR system to evolve from a controlled environment dictation task to a demanding BN task.
- Chapter 4 explains the work done in the development of our APP system. A detailed descriptions if given for all modules that compose the system and its evaluation. This chapter also covers the work done comparing our AP system with other state of the art algorithms.
- Chapter 5 addresses the development and integration of these modules into a fully working prototype media monitoring system.
- Finally, Chapter 6 presents the conclusions and future directions.

Chapter 2

BN Corpora

This chapter describes the speech and text resources used in this thesis. To support the research and development associated with a BN task it was necessary to collect representative European Portuguese speech and text resources in terms of amount, characteristics and diversity. These corpora served as fundamental resources for the development of all the modules in our media monitoring system, namely the audio pre-processing (APP), automatic speech recognition (ASR) and topic detection (TD) modules.

These corpora are constituted mainly by news reports from two sources: television Broadcast News (BN) and web Text News (TN). The BN corpora collected during this work are mostly composed by news shows transmitted by the Portuguese public television broadcast company (RTP). They are called ALERT-SR, COST278-BN and SSNT-BN corpus. The COST278-BN also has data from other 8 European languages. These BN resources were used for developing and testing APP, ASR and TD models. The TN corpus collected during this work is called WEBNEWS-PT and is composed by web journal articles from several European Portuguese daily newspapers. The TN resources were used for building and testing ASR and TD models.

The collection of all these resources was done in several stages and was supported mainly by the European project ALERT [Neto et al., 2003a], by the European COS278 action [Vandecatseye et al., 2004] and by national founding FCT (POSI/PLP/47175/2002).

The next few sections describe in detail each of these speech and text corpus in this order, first the BN corpora, ALERT-SR, COST278-BN, SSNT-BN and latter the TN corpus, WEBNEWS-PT. The chapter ends with a summary which reviews the work done.

2.1 ALERT-SR corpus

The ALERT-SR (SR = Speech Recognition) was the first European Portuguese BN corpus collected. The collection process started in April 2000 and lasted until the end of 2001.

It is entirely constituted by European Portuguese BN shows transmissions from Portuguese RTP station. It was used to train APP models, acoustic and language models for our ASR system, TD models and to evaluate performance.

RTP as data provider was responsible to collect the data in their installations. The corpus annotation process was jointly done by RTP and our laboratory (L2F). It was made by specialized transcribers trained by us. The corpus transcriptions were made using the Transcriber tool [Barras et al., 1998, Barras et al., 2001] following the LDC Hub4 [LDC-Hub4, 2000] (Broadcast Speech) transcription conventions. Table A.2 of Apendix A summarizes the LDC focus conditions used to classify different BN audio conditions.

The first set of ALERT-SR corpus is called ALERT-SR.pilot set and was all manually annotated. The other sets were first automatically segmented and transcribed by an early version of our BN APP and ASR modules and then manually corrected by the annotators. The prior automatic segmentation and transcription produced an enormous speed up of the annotation process although the first APP and ASR modules still had a high error rate.

This corpus was divided into six different sets, two were used for training (pilot and train sets) and four sets were used for testing purposes (devel, eval, jeval and 11march sets).

The first stage of the ALERT-SR corpus development was to collect a relative small pilot data set with approximately 5 hours including both audio and video. This pilot set was used to setup the collection process and discuss the most appropriate kind of shows to record. After the pilot set we began the collection and annotation process for the train and devel sets followed by the eval set. Later, two more evaluation sets were added: the jeval ("joint evaluation") set and the 11march ("11th march") set. Table 2.1 gives an overview of the ALERT-SR corpus in terms of quantity, duration and purpose of the data sets.

ALERT-SR set	News shows	Audio	Purpose
pilot	11	5 h	Training (cross-validation)
train	99	46 h	Training
devel	13	6 h	Development (tune parameters)
eval	12	4 h	ASR Evaluation
jeval	14	13 h	Media monitoring evaluation
11march	7	5 h	Evaluate daily LM adaptation
Total	156	79 h	

Table 2.1: ALERT-SR sets

As can be seen in Table 2.1 ALERT-SR has roughly 51 hours for training purposes, 6 hours for tuning parameters and 22 hours for evaluation purposes. The next subsections describe in detail each of these data sets.

2.1.1 ALERT-SR.pilot set

The ALERT-SR.pilot set was collected in April 2000 and served as a test bed for the capture and transcription processes. In this work it was used as cross-validation set for training the ASR acoustic models (phonetic classifiers).

We started by defining the type of programs to monitor, in close cooperation with RTP, being selected as primary goals all the news programs, national and regional, from morning to late evening, including both normal broadcasts and specific ones dedicated to sports and financial news. Given its broader scope and larger audience, the 8 o'clock evening news program was selected as the prime target. We selected one program of each type

(Table 2.2) resulting in a total of 11 shows with a duration of 5h 33m. After removing the jingles and commercial breaks we ended up with a net duration of approximately 5 h.

This set was manually transcribed at RTP. It includes the MPEG-1 video files (.mpeg) where the audio stream was recorded at 44.1 kHz at 16 bits/sample, a separated audio file (.wav) extracted from the MPEG-1 data, a transcription file (.trs) resulting from the manual annotation process with the Transcriber tool.

News show	Number	Duration	Type
Notícias	1	8 min	Morning news
Jornal da Tarde	1	47 min	Lunch time news
País Regiões	1	15 min	Afternoon news
País Regiões Lisboa	1	23 min	Local news
Telejornal	1	42 min	8 o'clock evening news
Remate	1	7 min	Daily sports news
24 Horas	1	21 min	Late night news
RTP Economia	1	8 min	Financial news
Acontece	1	17 min	Cultural news
Jornal 2	1	44 min	Evening news
Grande Entrevista	1	58 min	Political interview
Total	11	5 h	

Table 2.2: ALERT-SR.pilot set shows

Table 2.2 gives a brief summary of this set. The news shows collected have very different characteristics in terms of duration (some shows are only a few minutes long while other last almost one hour) and in terms of content (ranging from generic news, sports, cultural to political interview).

As one can see from the inspection of Figure 2.1 which shows the percentage of time according to the different LDC focus conditions (see a complete explanation in Appendix A), the ALERT-SR.pilot set has a high percentage of speech in the presence of background noise (F40 + F41). Also spontaneous speech dominates (F1 + F41) with 80% of the data. This can be accounted by the long political interview show and unfortunately by annotation mistakes. Figure 2.2 shows the distribution of sentences, from a total of 3234 sentences, by gender and focus conditions.

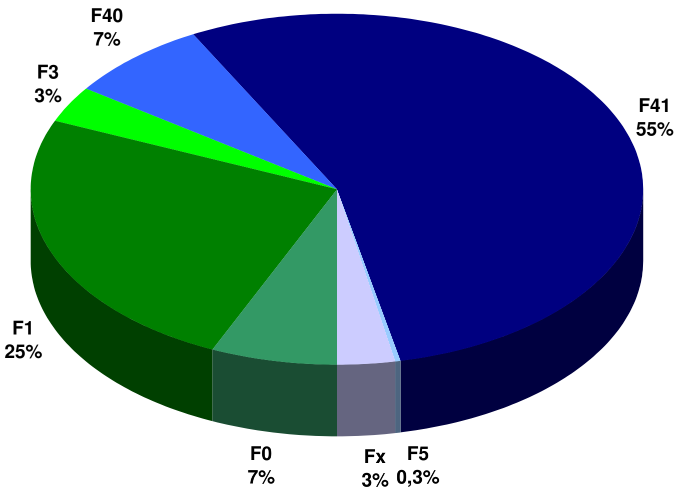

Figure 2.1: ALERT-SR.pilot set focus conditions time distribution

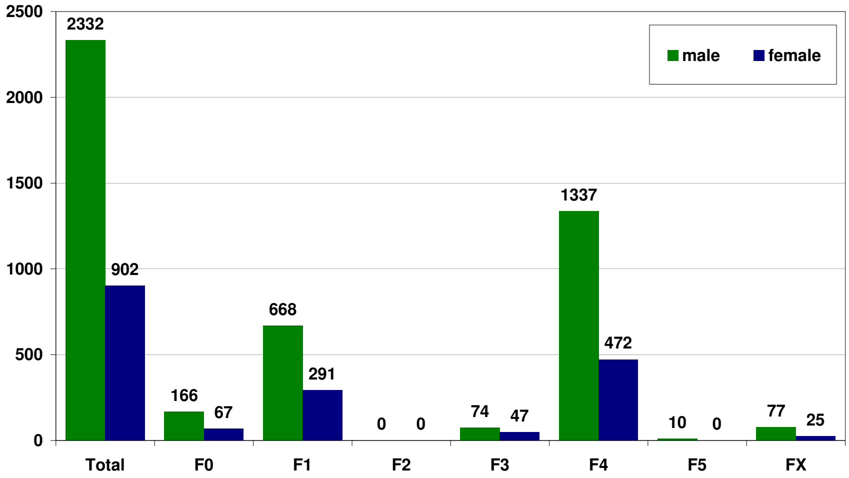

Figure 2.2: ALERT-SR.pilot set focus conditions gender sentences distribution

This set is not well balanced in terms of gender distribution with male speech being predominant having more than two thirds of the total number of sentences.

2.1.2 ALERT-SR.train set

This is ALERT-SR corpus main training set. In this work it was used for training APP models and ASR acoustic models (phonetic classifiers). The ALERT-SR.train set was recorded during one full month from 2000/10/09 to 2000/11/09. This set was jointly annotated by RTP and by our laboratory. The annotators were responsible for correcting the data that had been prior automatically segmented and recognized using APP and ASR modules already trained with BN data from the pilot set. This procedure resulted in a much faster annotation than doing everything manually. Nevertheless the annotation of all the training set which was done by 7 annotators took more than one year to complete.

This training set has only small changes in the type of shows collected when compared with the ALERT-SR.pilot set. It is constituted by 99 news shows with a net total duration of 46 h. The same base configuration was adopted as for the ALERT-SR.pilot set, except that we did not collected video streams. Also the audio was recorded at 32 kHz, due to restrictions of the hardware, and later downsampled to 16 kHz which was appropriate to the intended processing. It includes only a audio stream file (.wav) and a transcription file (.trs). Table 2.3 gives a brief summary of this set. The 8 o'clock evening news shows dominate, having more than half of the total duration. This was intentional since this is the main news show and is representative of the type of news we wanted to process.

News show	Number	Duration	Type
Notícias	8	1 h	Morning news
Jornal da Tarde	8	6 h	Lunch time news
País Regiões	13	5 h	Afternoon news
País Regiões Lisboa	7	2 h	Local news
Telejornal	30	24 h	8 o'clock evening news
24 Horas	4	1 h	Late night news
RTP Economia	13	1 h	Financial news
Acontece	9	2 h	Cultural news
Jornal 2	7	4 h	Evening news
Total	99	46 h	

Table 2.3: ALERT-SR.train set shows

From Figure 2.3 which shows the time distribution according to the different LDC focus conditions, we see that the ALERT-SR.train set is more balanced between clean (F0 + F1 = 37%) and noisy speech conditions (F40 + F41 = 53%) than the pilot set. Again in contrast with the pilot set we have here a better balance between planned speech (F0 + F40 = 57%) and spontaneous speech (F1 + F41 = 33%).

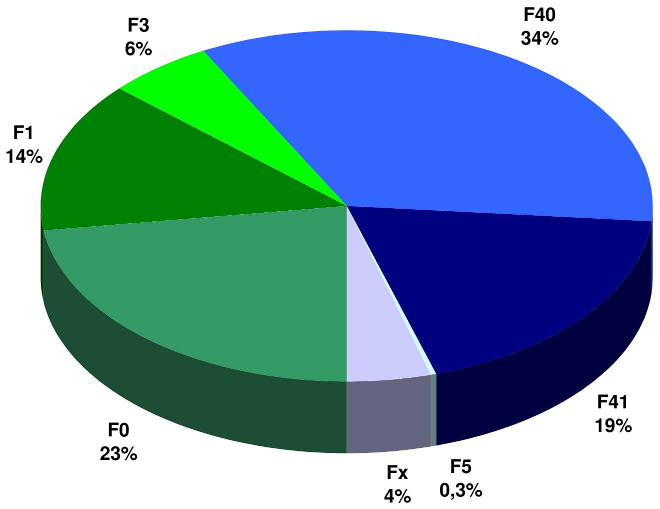

Figure 2.3: ALERT-SR.train set focus conditions time distribution

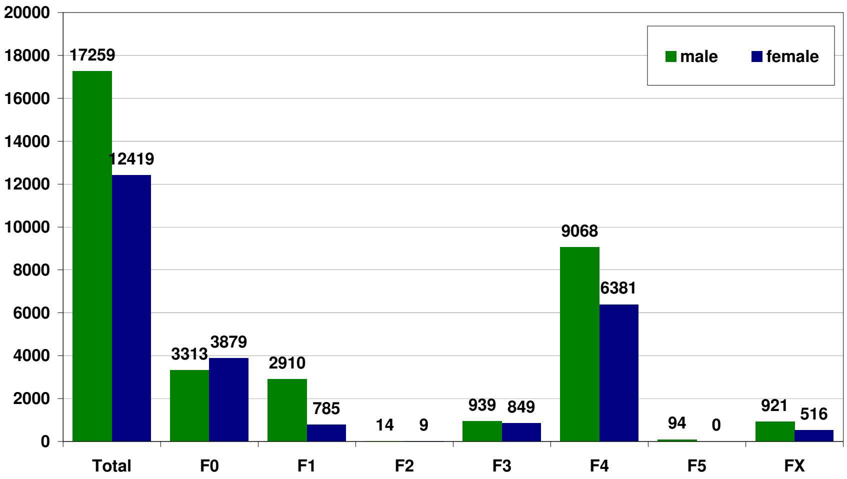

Figure 2.4: ALERT-SR.train set focus conditions gender sentences distribution

Figure 2.4 summarizes the distribution of sentences according to gender and focus conditions. Again we have here a much better balance between male and female sentences although male still has more data.

2.1.3 ALERT-SR.devel set

The purpose of the development set is to provide test data where free parameters of the system can be adjusted to maximize its performance. These parameters include for instance, threshold values for audio pre-processor modules and language models interpolation factors. This data set was recorded almost one month after the ALERT-SR.train set in the week of 2000/12/04 through 2000/12/10. It is composed by 13 news shows with a total useful duration around 6 h. Table 2.4 gives a brief summary of this test set.

News show	Number	Duration	Type
Notícias	1	9 min	Morning news
Jornal da Tarde	1	50 min	Lunch time news
País Regiões	1	23 min	Afternoon news
País Regiões Lisboa	1	18 min	Local news
Telejornal	3	175 min	8 o'clock evening news
24 Horas	2	51 min	Late night news
RTP Economia	2	7 min	Financial news
Acontece	1	14 min	Cultural news
Jornal 2	1	35 min	Evening news
Total	13	6 h	

Table 2.4: ALERT-SR.devel set shows

As we can observe its composition is very similar to the training set with the 8 o'clock news shows having aproximately half of the set total duration.

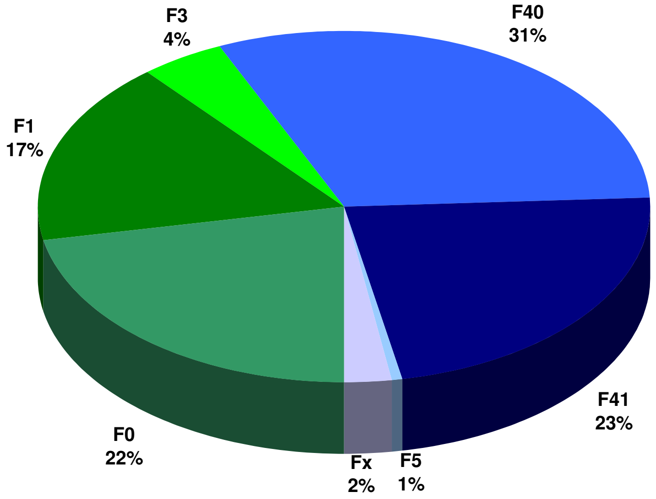

Figure 2.5: ALERT-SR.devel set focus conditions time distribution

From the inspection of Figure 2.5 which represents the development set percentage of speech according to focus conditions we see that the ALERT-SR.devel set is much similar to the training set with more or less the same time distribution. Here we have 39% of clean speech (F0 + F1) and 54% of noisy speech conditions (F40 + F41). In terms of planned/spontaneous speech here we have 54% of planned speech (F0 + F40) and 40% of spontaneous speech (F1 + F41).

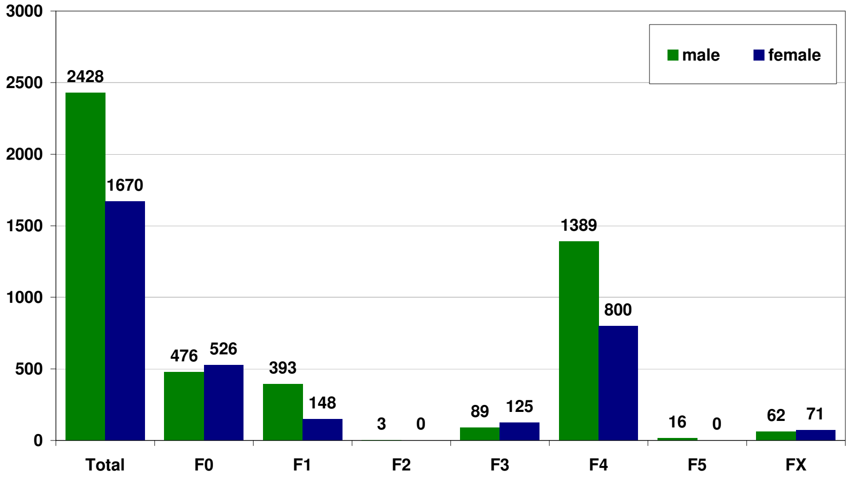

Figure 2.6: ALERT-SR.devel set focus conditions gender sentences distribution

Figure 2.6 summarizes the distribution of sentences according to gender and focus conditions. There are more male sentences than female when compared with the training set.

2.1.4 ALERT-SR.eval set

The purpose of the evaluation data set is to assess the ASR performance. It was recorded in the week of 2001/01/08 through 2001/01/15, almost one month after the development test set. It is composed by 12 news shows with a total useful duration over 4 h as reported in Table 2.5.

Figure 2.7 summarizes the evaluation set percentage of speech according to focus conditions. From the inspection of Figure 2.7 we see that the ALERT-SR.eval set has an enormous proportion of noisy conditions (F40 + F41 = 72%) compared with clean ones (F0 + F1 = 20%). The balance between planned and spontaneous speech is better with (F0 + F40 = 49%) planned and (F1 + F41 = 43%) spontaneous speech. Figure 2.8 summarizes

News show	Number	Duration	Type
Notícias	1	8 min	Morning news
Jornal da Tarde	1	43 min	Lunch time news
País Regiões	1	25 min	Afternoon news
País Regiões Lisboa	1	17 min	Local news
Telejornal	2	91 min	8 o'clock evening news
24 Horas	2	32 min	Late night news
RTP Economia	2	14 min	Financial news
Acontece	1	12 min	Cultural news
Jornal 2	1	26 min	Evening news
Total	12	4 h	

Table 2.5: ALERT-SR.eval test set composition

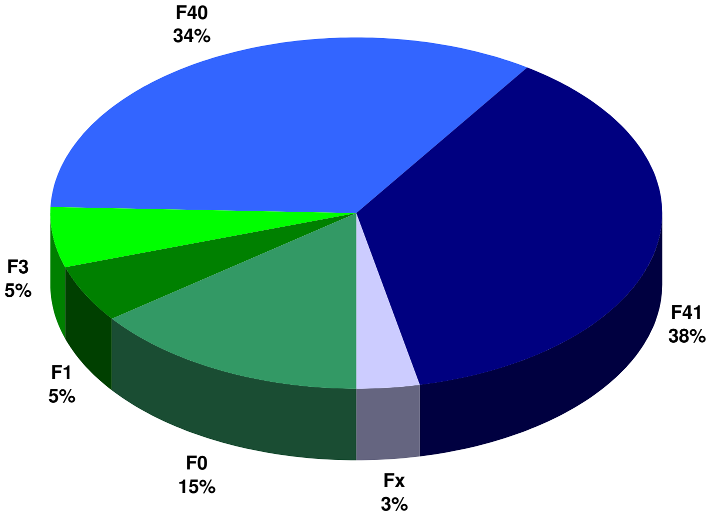

Figure 2.7: ALERT-SR.eval set focus conditions time distribution

the distribution of sentences according to gender and focus conditions. This set exhibits similar characteristics to the pilot set where male speech dominates. In this case it has more than double the number of sentences.

2.1.5 ALERT-SR.jeval set

The ALERT-SR.jeval ("joint evaluation") data set spans over two weeks from 2001/10/08 to 2001/10/21. Its original purpose was to evaluate the performance of the different topic detection algorithms developed by the partners of the ALERT project. Since these topic

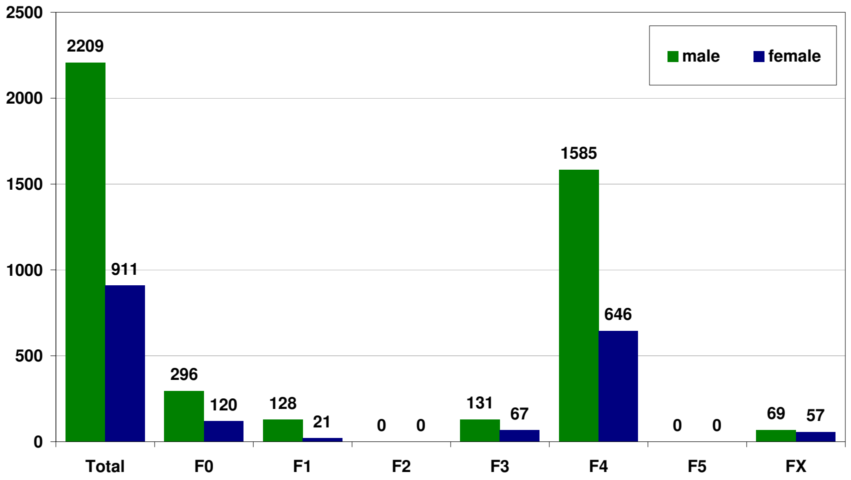

Figure 2.8: ALERT-SR.eval set focus conditions gender sentences distribution

detection algorithms worked for different languages (European Portuguese, French and German) an appropriate way to compare their performance was to have news shows with similar topic subjects. The joint evaluation data set covers the start of the Afghanistan war in 2001 which was reported simultaneously by Portuguese, French and German news shows. Although its original purpose the joint evaluation data is used as test set for assessing our media monitoring system performance, that is, to report the performance of APP, ASR and TD modules and to evaluate the error influence of prior modules in latter ones.

News show	Number	Duration	Type
Telejornal	14	13 h	8 o'clock evening news
Total	14	13 h	

Table 2.6: ALERT-SR.jeval set shows

It is composed only by 14 8 o'clock evening news shows with a total useful duration of 13 h as reported in Table 2.6. Figure 2.9 summarizes the joint evaluation set percentage of speech according to focus conditions.

From the inspection of Figure 2.9 we see that the ALERT-SR.jeval has much more speech in noisy conditions (F40 + F41 = 66%) than clean speech (F0 + F1 = 29%). Planned speech dominates with F0 + F40 = 63% and spontaneous speech only F1 + F41 = 32%.

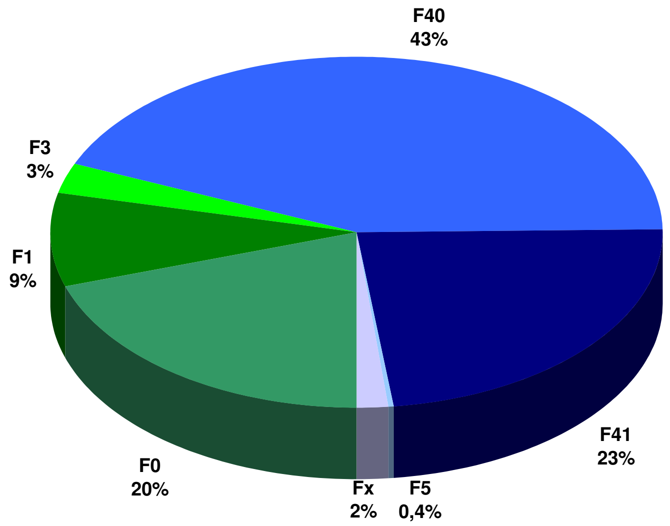

Figure 2.9: ALERT-SR.jeval set focus conditions time distribution

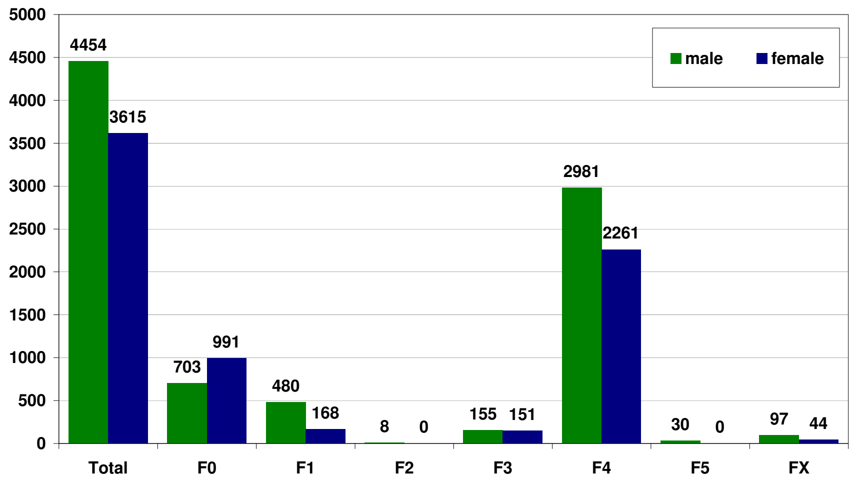

Figure 2.10: ALERT-SR.jeval set focus conditions gender sentences distribution

Figure 2.10 summarizes the distribution of sentences according to gender and focus conditions. This set has a good balance between male and female speech specially when compared with other sets from the ALERT-SR corpus.

2.1.6 ALERT-SR.11march set

The ALERT-SR 11th March data set spans over one week from 2004/03/08 to 2004/03/14. It is composed by 7 news shows and a total useful duration of 5 hours. The purpose of this test set was to evaluate algorithms for the daily vocabulary adaptation of ASR language model. This test set is specially appropriate for this since it covers at the middle of the week the terrorist attacks in Madrid where new (out of ASR vocabulary) words are introduced. Table 2.7 gives a brief summary of this test set.

News show	Number	Duration	Type
Telejornal	7	5 h	8 o'clock evening news
Total	7	5 h	

Table 2.7: ALERT-SR.11march test set shows

Figure 2.11 summarizes the "11th march" set percentage of speech according to focus conditions.

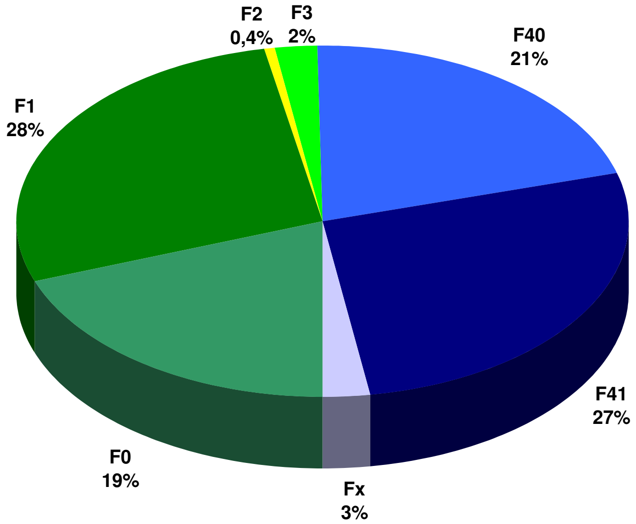

Figure 2.11: ALERT-SR.11march focus conditions time distribution

As we can see from the inspection of Figure 2.11 this is a more balanced set. Clean speech (F0 + F1 = 47%) has almost the same amount of time than noisy speech (F40 + F41 = 48%). Planned speech (F0 + F40 = 40%) is less than spontaneous speech (F1 + F41 = 55%) which can be explained by live reports from the terrorist attacks scenery.

Figure 2.12 summarizes the distribution of sentences according to gender and focus conditions. Gender is very badly distributed. Female speakers have more noisy speech sentences and male speakers which dominate overall have much more clean speech sentence.

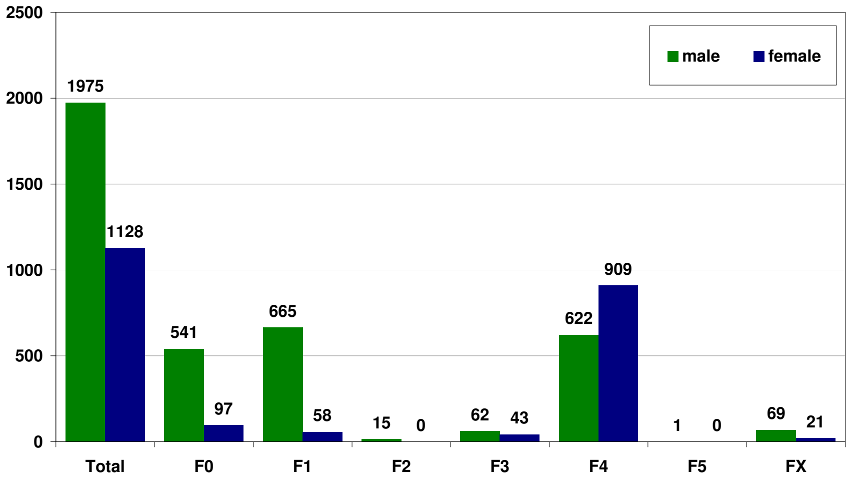

Figure 2.12: ALERT-SR.11march focus conditions gender sentences distribution

A possible explanation for this is because we have female field reporters and male (in studio) commentators and political analysts.

2.2 COST278-BN corpus

The pan-European COST278-BN database was collected by a BN special interest group within the European Union COST action number 278. The main goal of this corpus was to provide appropriate data for the evaluation of language independent algorithms such as audio pre-processing algorithms. The objective was not to create a large database for the training of complete transcription systems, but rather a modest database for accommodating system adaptation, language detection and development of language independent parts such as audio pre-processing modules. By evaluating different algorithms using the same multilingual/ multi-style corpus, it may be possible to assess the strengths and weaknesses of existing approaches, and to conceive better approaches through collaborative research. This common evaluation data enabled us and a number of European research groups to compare directly the performance of our (different) algorithms.

Each institution collected a national data set consisting of a number of complete news broadcasts from public and/or private TV stations. The goal was to collect approximately 3 hours of material per language data set. At present the COST278-BN corpus consists of 30 hours of Broadcast News recordings, divided into 10 equally large national data sets. Each national set contains some complete news shows broadcasted by TV stations in one country or region. Each national set was recorded and transcribed by one institution and was performed according to a protocol described in [Vandecatseye et al., 2004]. Since two institutions from Slovenia participated in the data collection, the database covers nine European languages: Belgian Dutch (BE), Portuguese (PT), Galician (GA), Czech (CZ), Slovenian (SI and SI2), Slovak (SK), Greek (GR), Croatian (HR) and Hungarian (HU).

Language	News shows	Duration	Anchors	TV stations
BE	6	3 h	8	VRT
CZ	10	3 h	12	Prima, Nova, CT1
GA	3	3 h	3	TVG
GR	3	3 h	5	NET
HR	6	3 h	3	HRT
HU	11	3 h	6	MTV1, MTV2, RTL Klub
PT	6	3 h	6	RTP1, RTP2
SI	3	3 h	6	RTV-SLO1
SI2	3	3 h	4	RTV-SLO1
SK	9	3 h	7	TA3
Total	60	30 h	60	

Table 2.8: COST278-BN corpus news shows

Table 2.8 shows the names of the TV stations, the number of collected shows, the number of different anchor persons appearing in these shows and durations. Due to the limited size of each national sets (around 3 hours) this data is not appropriate for training ASR algorithms, but it is suitable for evaluating audio pre-processing algorithms and possibly evaluating adaptation methods for speech recognition. A particular property of the COST278 database is that it also includes the video files. They can help to check the correctness of speaker labels, and support the development of multimedia and multi-modal algorithms. To save memory, the video data were archived in Real Media Video format with a resolution of 352x288. Figure 2.13 summarizes the COST278-BN database percentage of speech according to focus conditions.

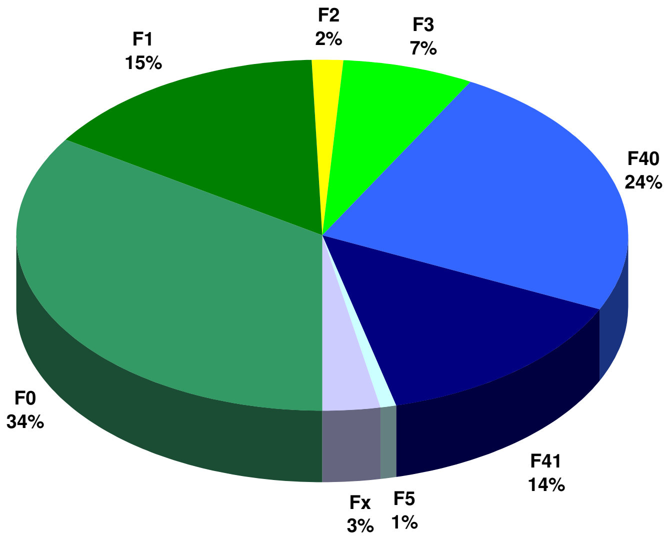

Figure 2.13: COST278-BN focus conditions time distribution

From the inspection of Figure 2.13 we see that this database has 49% of clean speech (F0 + F1) and a lesser percentage of noisy speech (F40 + F41 = 38%) than ALERT-SR sets. Furthermore, this database exhibits measurable quantities of telephone speech (F2 = 2%) and non-native speech (F5 = 1%). Planned speech (F0 + F40 = 58%) clearly dominate spontaneous speech (F1 + F41 = 29%). Additionally, this database has a high percentage of speech over music (F3 = 7%). Figure 2.14 summarizes the distribution of sentences according to gender and focus conditions. Again, here we observe a male speech dominancy.

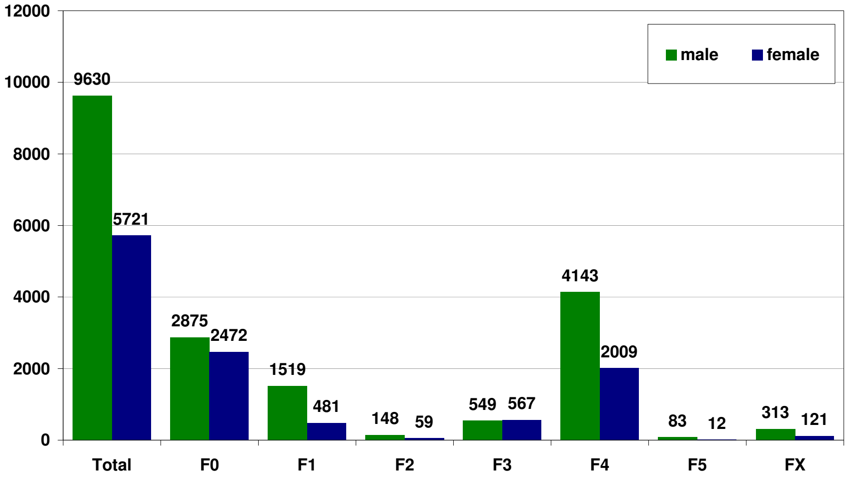

Figure 2.14: COST278-BN focus conditions gender sentences distribution

2.3 SSNT-BN daily news show database

Building accurate BN statistic models for APP and ASR requires large quantities of training data. More speech data means better representative statistics and consequently models will be more accurate. Current state of the art ASR systems use always more than 200 hours of carefully annotated training data [Gales et al., 2007]. The most recent ones, that participated in NIST RT04 Fall used more than 2000 hours of speech [NIST, 2004]. Our main BN speech corpus, ALERT-SR, has a limited amount of training data, around 46 hours, which is insufficient if we want to improve the ASR system performance. We will need more training data to build for instance gender dependent acoustic models. Careful collection and annotation of speech data is a very expensive process both in terms of time and money. The solution to this problem is to automatically collect and transcribe data and then have some unsupervised selection process using confidence measures to select the most accurate speech portions and add them to the training set.

With this purpose in mind, since March 2003 we started collecting and automatically processing on a daily basis the 8 o'clock news show from RTP station with our media monitoring prototype system. Until the end of 2006 this huge speech database had over 180 giga bytes of data. So far we have reused some of this speech material and have formed 3 data sets called: "one-month-2005", "jornal2" and "six-months-2006". The first data set contains one complete month of "Telejornal" news shows from 2005. The second data set has 52 news shows from the 9 o'clock news show of RTP2 station which were collected by our prototype from 2004/03/12 until 2004/05/17. The third data set represents six complete months of 8 o'clock news shows collected during 2006.

The data annotation was performed in several stages. Everything was processed automatically by the modules that compose our Media monitoring prototype (for a full explanation and understanding of these modules please consult Chapters 3, 4 and 5). In the "one-month-2005" and "jornal2" data sets the jingle detection (JD) module results were manually inspected and corrected to prevent false alarms. After this jingle detection validation the data was automatically processed by our APP and ASR modules to determine useful speech parts, transcribe them and generate for each word a confidence level. The words that have a confidence value higher than 91.5% are used for generating phonetic labels (called desired outputs) appropriate for training the MLP phonetic classifiers. The words having low confidence scores were marked as "don't use in training". After this automatic annotation process the new 3 sets represented a considerable amount of training data. Table 2.9 gives a brief summary of these three new data sets.

SSNT-BN data sets	News shows	Durations		
		After JD	After APP	After ASR
one-month-2005	30	32 h	27 h	17 h
jornal2	52	28 h	25 h	16 h
six-month-2006	178	166 h	142 h	100 h
Total (3 sets)	260	226 h	194 h	133 h

Table 2.9: Statistics from the 3 data sets of SSNT-BN corpus

We obtain approximately 17 hours of useful speech per month from the 8 o'clock news shows ("Telejornal"). Extrapolating to the totality of SSNT-BN database (45 months) it gives an estimate of over 700 hours of useful speech that can be added to our current BN training set. Figure 2.11 shows the "jornal2" data set time distribution of speech according to the APP background classification (clean, music, noise). From the inspection of Figure 2.15 we see that this data set has 41.7% of clean speech and a higher percentage of noisy conditions. This is similar to what was observed in the manual annotations of all our BN corpus. Figure 2.16 summarizes the distribution of sentences according to automatic APP gender and background conditions. Again, as in the other BN speech databases this set has much more male than female speech sentences.

The other two sets ("one-month-2005" and "six-months-2006") have similar statistics to what was observed in the ALERT-SR corpus.

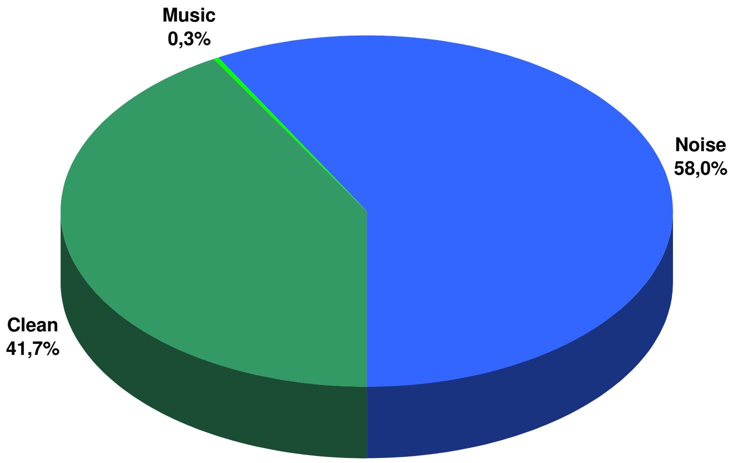

Figure 2.15: SSNT-BN.jornal2 set focus conditions time distribution

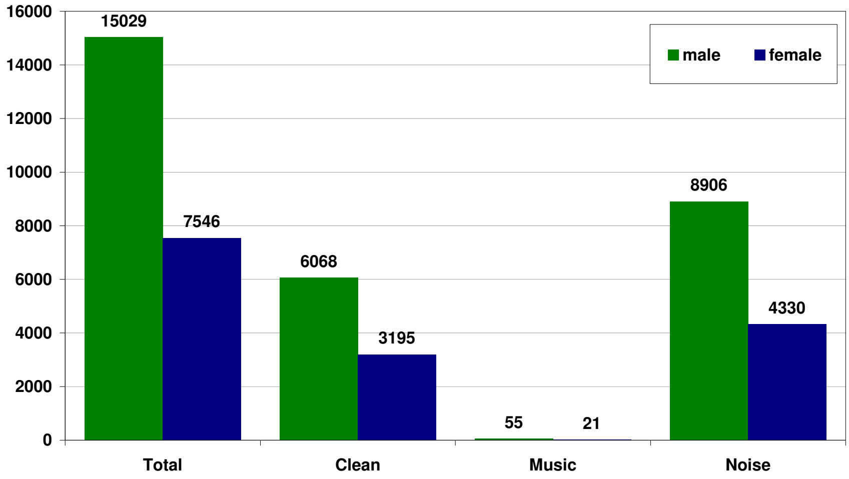

Figure 2.16: SSNT-BN.jornal2 set focus conditions gender sentences distribution

2.4 WEBNEWS-PT Text Newspaper corpus

Statistical ASR language models require large quantities of text for a correct estimation of the n-grams. Given that we are building an ASR system for a BN task the type of texts used have to be appropriate. Preferably it should be newspaper texts and transcriptions from BN news shows. Since 2001 we started to collect every day the online editions of all the major Portuguese newspapers. This Text Newspaper corpus is called WEBNEWS-PT and is mainly used to train ASR statistical n-gram language models. Currently (until the end of 2005) the WEBNEWS-PT corpus has texts collected from different newspapers styles (daily newspapers covering all topics, weekly newspapers also with a broad coverage of topics, economics newspapers and sports news). This corpus has over 42 million sentences and 740 million words. Table 2.10 gives a brief summary of this TN corpus.

Newspaper	Sentences	Words	Type
A Bola	1.9 M	32.2 M	Daily, sports
Diário de Notícias	5.2 M	88.9 M	Daily, generic
Diário Económico	5.9 M	66.6 M	Daily, economics
Expresso	2.0 M	39.9 M	Weekly, generic
Jornal de Notícias	5.4 M	94.5 M	Daily, generic
O Jogo	6.9 M	91.0 M	Daily, sports
O Independente	0.2 M	2.4 M	Weekly, generic
O Público	14.9 M	325.8 M	Daily, generic
Total	42.4 M	741.3 M	

Table 2.10: WEBNEWS-PT composition

Each newspaper web edition is automatically collected using specially tailored scripts (web crawlers). There are huge differences amongst web editions and the tailored scripts reduce the time and space taken to collect the edition by storing only the relevant html pages (the ones with the news articles). This also has the advantage of preserving important information like the topic of the article which can latter be used for creating topic LM or TD models. Unfortunately web newspaper change the layout style too often (possibly to give an image of dynamism or to captivate more audience). This is the biggest disadvantage of having tailored scripts appropriate for each newspaper since their maintenance becomes more difficult.

After collecting the daily web edition of a given newspaper the scripts convert the text from html format into simple text format. Again special conversion scripts were made to suit each newspaper edition. At this stage the processing scripts check to see if there are repeated news articles by comparing each article with last day articles from the same newspaper. Afterwards the simple text is normalized (expansion of abbreviations and contractions, numbers and digits to text, etc). The resulting normalized sentences are then sgml tagged and compressed to save disk space. The text is now ready for being used.

2.5 Summary

Due to its complexity BN tasks need enormous amounts of data both acoustic and text for building accurate models. For the European Portuguese language no BN speech resources were available and new corpora had to be collected. The collection of these corpora represented a huge effort in terms of time. The author and the colleague Rui Amaral devoted many hours of work in their definition and collection, training of annotators and validation of data.

The corpora described in this chapter were fundamental resources for the development of our BN media monitoring system. The ALERT-SR corpus with its 6 data sets is the most important European Portuguese BN database. It was used for training APP and ASR models and for evaluating ASR and TD performance. The COST278-BN corpus is first of all useful for the evaluation of APP algorithms and for permitting the direct comparison of results obtained by different algorithms developed by several different groups. By joining forces with some European partners it was possible to construct a modest multilingual/multi-style BN database of audio and video files. The SSNT-BN corpus is a work in progress. It has a huge potential for producing future training material because of its size and the new acoustic models trained so far (see Chapter 3) have demonstrated gains in recognition performance. The same can be said for our TN corpus, the WEBNEWS-PT database. It is increasing in size every day.

Chapter 3

Automatic Speech Recognition

This chapter gives a complete overview of our Automatic Speech Recognition (ASR) core, AUDIMUS and a detailed description of the work done to improve its functionality and performance. Through the participation in several projects, national and international, our research laboratory (L2F) acquired over the years a vast experience in developing ASR systems for the English language with a major focus in speaker-adaptation techniques [Neto et al., 1996, Neto, 1998]. Since then we have been using that experience to develop ASR systems for the European Portuguese language. Currently our ASR core, AUDIMUS, has been ported successfully to several different tasks like dictation, telephone, Broadcast News, and portable devices just to name the most important ones. This chapter starts by reviewing the development of AUDIMUS and then looks into the details of AUDIMUS.media an implementation suitable for the BN task. Finally we review the modifications done in AUDIMUS.media in order to improve the overall performance.

3.1 AUDIMUS ASR system

AUDIMUS was initially developed for the European Portuguese language as a large vocabulary continuous ASR system for a dictation tasks [Neto et al., 1997a, Neto et al., 1998].

It was developed using a new European Portuguese speech corpus the BD-PUBLICO database [Neto et al., 1997b], collected during this author MSc. thesis and similar in characteristics to the Wall Street Journal database [Paul and Baker, 1992]. Figure 3.1 represents AUDIMUS system block diagram.

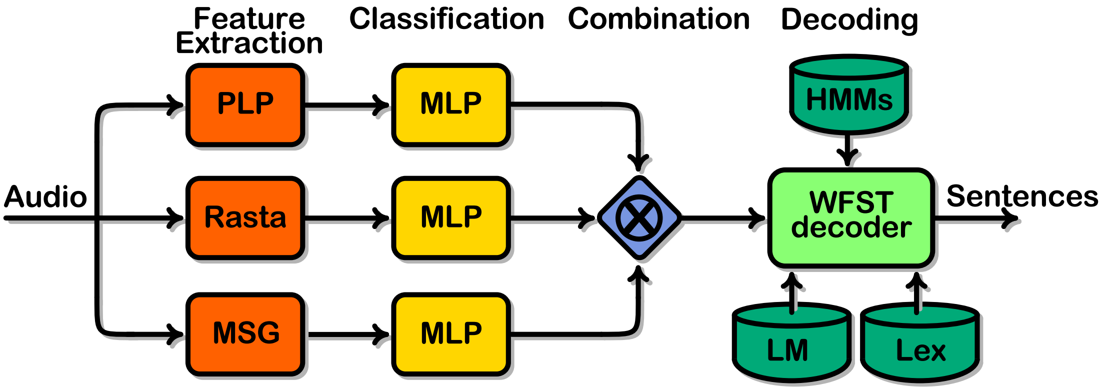

Figure 3.1: AUDIMUS system overview.

The next subsections give a brief overview of AUDIMUS constituent parts: Acoustic Modelling encompassing Feature Extraction, Phonetic Classification and Combination and the Decoding which encompasses the decoder algorithm, the Hidden Markov Models, the Language Model and the Lexicon.

3.1.1 Acoustic Modelling

AUDIMUS acoustic model combines the temporal modelling capabilities of Hidden Markov Models (HMMs) with the pattern discriminative classification capabilities of multilayer perceptrons (MLPs) [Bourlard and Morgan, 1994]. In this hybrid HMM/MLP system a Markov process is used to model the basic temporal nature of the speech signal. The MLP is used as the acoustic model estimating context-independent posterior phone probabilities given the acoustic data at each frame. The acoustic modelling of AUDIMUS combines phone probabilities generated by several MLPs trained on distinct feature sets resulting from different feature extraction processes. These probabilities are taken at the output of each MLP classifier and combined using an average in the log-probability domain [Meinedo and Neto, 2000]. All MLPs are gender independent and use the same phone set constituted by 38 context independent phones for the Portuguese language plus silence (see Table B.1 in Appendix B). The combination algorithm merges together the probabilities associated to the same phone.

AUDIMUS uses three different feature extraction methods and MLPs with almost the same basic structure, that is, an input layer with 7 or 9 context frames, a non-linear hidden layer with 500 sigmoidal units and 39 softmax [1] outputs. The feature extraction methods are PLP, Log-RASTA [Hermansky et al., 1992] and MSG [Kingsbury et al., 1998]. The features are extracted from the incoming audio signal using a sliding window of 20 ms, which is updated every 10 ms. The PLP and Log-RASTA features extract 26 parameters per frame (12th order coefficients plus log energy plus first order derivatives). The MSG extracts 28 features per frame of coefficients. The MLPs for PLP and Log-RASTA have exactly the same architecture, schematically (7x26)-500-39. The MSG features MLP has a slightly larger context window, (9x28)-500-39. Experiments in the BD-PUBLICO development test set showed this was the best architecture.

3.1.2 Lexicon

The lexicon for the dictation task was built by selecting the vocabulary of the training and development sets from the BD-PUBLICO database. The resulting list had 27 k different words and was then phonetically transcribed by a rule grapheme to phone system generating an initial set of pronunciations. This automatically generated lexicon was then hand revised by a specialised linguist generating a multi pronunciation lexicon with 32,148 pronunciations.

[1] The outputs are normalized so that their sum equals one.

3.1.3 Language Model

The language model built for AUDIMUS was a 3-gram backed-off model created from 3 years of ”O PÚBLICO” newspaper texts amounting to 46 M words. Smoothing was done by absolute discounting. This language model had around 6 M probabilities and had a perplexity of 107 in the BD-PUBLICO database development test set [Neto et al., 1998].

3.1.4 Decoder

AUDIMUS used a decoder with an efficient search strategy based on A* algorithm and on stacks [Renals and Hochberg, 1995b, Renals and Hochberg, 1995a, Renals and Hochberg, 1999]. This search algorithm uses posterior phone probability estimates generated by the MLPs acoustic model. This decoder implements an efficient method of reducing the search space and thus the computational effort by pruning the total number of hypothesis to consider. This pruning method is called phone deactivation pruning [Renals and Hochberg, 1995b].

3.1.5 Results

After 7 iterations of realignment and training of the acoustic model, this ASR system was evaluated in the BD-PUBLICO evaluation test set showing 16.7% of WER [Neto et al., 1998].

3.2 Errors in a BN task

Performance for ASR systems is strongly dependent on many factors such as system limitations (limited vocabulary, trained for the speaker or not), speech problems (accent, style, spontaneous, disfluencies), channel problems (background noises, low bandwidth such as telephone). In a Broadcast News task all these limiting factors are present (many times simultaneously) and can have a significant influence in the performance of the ASR system. BN speech data is normally very complex from the acoustical and language point of view covering spontaneous speech, hesitations, many different types of background noise, music, channel conditions. A qualitative analysis indicates the following types of errors present in BN speech:

- Errors due to severe vowel reduction. Vowel reduction, including quality change, devoicing and deletion, is specially important for European Portuguese, being one of the features that distinguishes it from Brazilian Portuguese and that makes it more difficult to learn for a foreign speaker. It may take the form of (1) intra-word vowel devoicing; (2) voicing assimilation; and (3) vowel and consonant deletion and coalescence. Both (2) and (3) may occur within and across word boundaries. Contractions are very common, with both partial or full syllable truncation and vowel coalescence. As a result of vowel deletion, rather complex consonant clusters can be formed across word boundaries. Even simple cases, such as the coalescence of the two plosives (e.g. *que conhecem*, ’who know’), raise interesting problems of whether they may be adequately modelled by a single acoustic model for the plosive. This type of error is strongly dependent on factors such as high speech rate. The relatively high deletion rate may be partly attributed to severe vowel reduction and affects mostly (typically short) function words.

- Errors due to OOVs. This affects namely foreign names. It is known that one OOV term can lead to between 1.6 and 2 additional word errors [Gauvain et al., 1995].

- Errors in inflected forms. This affects mostly verbal forms (Portuguese verbs typically have above 50 different forms, excluding clitics), and gender and number distinctions in names and adjectives. It is worth exploring the possibility of using some post-processing parsing step for detecting and hopefully correcting some of these agreement errors. Some of these errors are due to the fact that the correct inflected forms are not included in the lexicon.

- Errors around speech disfluencies. This is the type of error that is most specific of the spontaneous speech, a condition that is fairly frequent in our BN corpora. The frequency of repetitions, repairs, restarts and filled pauses is very high in these conditions, in agreement with values of one disfluency every 20 words cited in [Shriberg, 2005]. Unfortunately, the training corpus for Broadcast News included a very small representation of such examples.
- Errors due to inconsistent spelling of the manual transcriptions. The most common inconsistencies occur for foreign names or consists of writing the same entries both as separate words and as a single word (”primeiro-ministro” and ”primeiro” ”ministro”).

3.3 AUDIMUS.media ASR system

The development of our BN ASR system, AUDIMUS.media was an iterative task. It evolved from the dictation version into a version suitable for BN by a large number of small steps. First, new vocabulary, lexicon and language model were built for the new task.

3.3.1 Vocabulary and pronunciation lexicon

To achieve a reasonable coverage of the European Portuguese language in a task with such a broad range of topics one would expect to need a larger vocabulary size than the 27 k words we used for the dictation task. Generally BN ASR systems developed for the English language are based on 64 k vocabulary size. We followed the same approach although the European Portuguese is more flexive than English. From the 335 M words of newspaper text that composed our newspaper text corpus at the time, 427 k different words were extracted. Around 100 k occur more than 50 times in the texts. These words were selected and classified according to syntactic classes. From that set of words a subset of 57 k was selected based on the weighted frequency of their occurrence in the text corpus according to the syntactic classes. This set was augmented with basically all words from the available transcripts of the ALERT-SR.train set (at the time containing 12,812 different words from a total of 142,547). The margin to a 64 k vocabulary would be used to incorporate the new words of the training data. From the vocabulary we were able to build the pronunciation lexicon. To obtain the pronunciations we used different lexica available in our laboratory. For the words not present in those lexica (mostly proper names, foreign names and some verbal forms) we used an automatic grapheme to phone system to generate corresponding pronunciations. Our final lexicon has a total of 65,895 different pronunciations. In ALERT-SR.devel test set which has 5,426 different words in a total of 32,319 words, the number of out of vocabulary words (OOVs) using our 57 k word vocabulary was 444 words representing a OOV word rate of 1.4%.

3.3.2 Language model

The first language model built for the new BN task was a backed-off 4-gram extracted from the 335 M words of the WEBNEWS-PT corpus. The smoothing technique used was absolute discounting [Meinedo et al., 2001]. Later, with more newspaper text available (around 384 M words) another better model was built [Souto et al., 2002].

A language model generated from newspaper texts does not perform as well as one would expect because sentences spoken in BN do not match the style of the sentences written in a newspaper. A language model built entirely from BN transcriptions would certainly be more adequate. Unfortunately, we do not have enough BN transcriptions to generate a satisfactory language model. However we can adapt better the language model built from newspaper texts to the BN task by combining it with a language model created from BN transcriptions using linear interpolation [Souto et al., 2002]. The language model generated from the available training set transcriptions of the ALERT-SR corpus (ALERT-SR.train and ALERT-SR.pilot sets) (around 142 k words) is a backed-off 3-gram model using absolute discounting. The optimal weights used in the interpolation were computed for the ALERT-SR.devel test set [Souto et al., 2002].

Since then all the language models used in our ASR BN system were interpolated models. Each time more data was available (newspaper texts or BN training transcriptions) a new better language model was created.

Recently [Martins et al., 2005], in one of these language model updates, a better type of smoothing was employed Kneser-Ney and also entropy pruning. These techniques produced a better LM and consequently yielded a performance increase in speech recognition. The perplexity obtained in a development set is 112.9.

3.3.3 Decoder

We switched from the NOWAY decoder used for the dictation task to a new decoding algorithm. The decoder underlying the AUDIMUS.media ASR system is based on the Weighted Finite-State Transducer (WFST) approach to large vocabulary speech recognition [Caseiro and Trancoso, 2000, Caseiro and Trancoso, 2002]. In this approach, the search space used by the decoder is a large WFST that maps observation distributions to words. This WFST consists of the composition of various transducers representing components such as: the acoustic model topology H ; context dependency C ; the lexicon L and the language model G. The search space is thus $H \circ C \circ L \circ G$ (we use the matrix notation for composition), and is traditionally compiled outside of the decoder, which uses it statically. Our approach differs in that our decoder is dynamic and builds the search space on-the-fly required by the particular utterance being decoded [Caseiro and Trancoso, 2001b]. Among the advantages provided by the dynamic construction of the search space are: a better scalability of the technique to large language models; reduction of the memory required in runtime; and easier adaptation of the components in runtime. The key to the dynamic construction of the search space is our WFST composition algorithm [Caseiro and Trancoso, 2001a, Caseiro and Trancoso, 2002] specially tailored for the integration of the lexicon with the language model (represented as WFSTs). Our algorithm performs simultaneous the composition and determinization of the lexicon and the language model while also approximating other optimizing operations such as weight pushing and minimization [Caseiro and Trancoso, 2003]. The goal of the determinization operation is to reduce lexical ambiguity in a more general way than what is achieved with the use of a tree-organized lexicon. Weight pushing allows the early use of language model information, which allows the use of tighter beams thus improving performance. Minimization essentially reduces the memory required for search while also giving a small speed improvement.

3.3.4 Confidence Measures

Associating confidence scores to the recognized text is essential for evaluating the impact of potential recognition errors. Hence, confidence scoring was recently integrated in the ASR module. In a first step the decoder is used to generate the best word and phone sequence including information about the word and phone boundaries as well as search space statistics. Then for each recognized phone a set of confidence features are extracted from the utterance and from the statistics collected during decoding. The phone confidence features are combined into word level confidence features. Finally a maximum entropy classifier is used to classify words as correct or incorrect. The word level confidence feature set includes various recognition scores (recognition score, acoustic score and word posterior probability [Williams, 1999]), search space statistics (number of competing hypotheses and number of competing phones) and phone log likelihood ratios between the hypothesized phone and the best competing one. All features are scaled to the $[0, 1]$ interval. The maximum entropy classifier [Berger et al., 1996] combines these features according to:

$$P(correct|w_i) = \frac{1}{Z(w_i)} exp[\sum_{j=1}^{F} \lambda_j f_j(w_i)] \tag{3.1}$$

where w_i is the word, F is the number of features, $f_j(w_i)$ is a feature, $Z(w_i)$ is a normalization factor and λ_j are the model parameters. The detector was trained using the ALERT-SR.train set. When evaluated on the ALERT-SR.jeval test set an Equal-Error-Rate (EER) of 24% was obtained.

3.3.5 Evaluation

All evaluations were conducted using the NIST toolkit *sclite* [NIST, 2000]. This software calculates the Word Error Rate (percentage of word errors) given a set of reference sentences and a corresponding set of hypothesis recognised sentences. Word Error Rate can be defined as the percentage of correctly recognised words subtracted of the insertions.

$$\text{WER} = \frac{\text{correct} - \text{inserted}}{\text{correct} + \text{missed}}$$

The other important evaluation factor is the time taken to recognise a set of hypothesis. Normally this is expressed in terms of times real time (xRT). If a sentence has 1 minute of audio and the ASR system takes 10 minutes to process it, that means the system works in 10 times real time. This factor is highly dependent on hardware and through the years computers have evolved substantially becoming orders of magnitude faster. It would not be fair to present xRT results because they would not be directly comparable. Nevertheless our current BN ASR system (denoted align 8 in Table 3.1) works in less than real time in a P4 dual core machine @ 2.8 GHz computer with 2 G bytes memory.

3.3.6 Results

Table 3.1 represents the recognition evaluation of all versions of AUDIMUS.media obtained through the years of development of our BN ASR system. The first 3 columns of Table 3.1 refer to the acoustic model. The first one indicates the alignment number, the second indicates the MLPs topology and the third one the number of training hours in the ALERT-SR.train set (which as we explained in Chapter 2 increased at the same time AUDIMUS.media was being developed). The fourth and fifth columns of Table 3.1 refer the language model and decoder used. Finally the last 3 rightmost columns indicate the number of test hours and the name of the test set used, the WER for the F0 focus condition and the WER for all focus conditions.

System	Acoustic model	Train Hrs	Language model	Decoder	Test Hrs	% WER F0	% WER All
AUDIMUS	7x26-500-39	—	newspaper, 335 Mw, 4g	Stack, A*	1 h (devel)	30.6	55.4
Align 1	”	5 h	”	”	”	23.2	42.1
Align 2	”	13 h	”	”	”	20.2	41.3
Align 3	”	23 h	newspaper, 384 Mw, 4g	”	3 h (devel)	21.4	41.5
Align 4a	7x26-1000-40	”	”	”	”	19.4	35.6
Align 4b	”	”	interpolated, 384 Mw + 142 kw, 4g	”	”	18.4	35.2
Align 5a	”	”	interpolated, 434 Mw + 142 kw, 4g	”	6 h (devel)	18.3	33.6
Align 5b	”	”	”	WFST	”	18.7	32.0
Align 6	”	46 h	”	”	”	18.8	31.6
Align 7a	7x26-2000-40	”	”	+ det min L	”	18.0	30.7
Align 7b	7x26-4000-40	”	”	”	”	16.9	29.1
Align 7c	”	”	”	+ eager prune	”	16.7	28.9
Align 7d	”	”	”	+ shorter HMMs	”	14.8	26.5
Align 7e	”	”	”	”	4 h (eval)	11.8	28.2
Align 7f	”	”	interpolated, 604 Mw + 532 kw, 4g	”	”	11.5	26.7
Align 7g	”	”	”	”	13 h (jeval)	15.8	26.4
Align 7h	”	”	”	”	5 h (11march)	14.6	28.1
Align 8a	11x26-1000-1000-40	”	”	”	4 h (eval)	10.2	24.0
Align 8b	13x26-1500-1500-40	”	”	”	”	10.0	23.7
Align 8c	”	”	”	”	13 h (jeval)	11.3	23.5

Table 3.1: BN speech recognition evaluation using ALERT-SR test sets

The first line of results in Table 3.1 is for the AUDIMUS baseline system whose acoustic model had been trained only on clean read speech from the BD-PUBLICO database. The MLPs had a topology of 7 input context frames of 26 feature coefficients, a single hidden layer with 500 units and an output layer with 39 phone classes, or schematically (7x26)-500-39. The 4-gram language model was estimated from 335 Mwords of newspaper texts only. AUDIMUS used the NOWAY (Stack, A* algorithms) and was tested using 1 hour of speech from the ALERT-SR.devel test set. Looking at the results the WER was very high even for the F0 focus condition which should in theory be more similar to the training conditions of AUDIMUS (dictation task). Although F0 speech is from clean prepared speech the TV professional speakers have a much higher speaking rate than the readers recorded in the BD-PUBLICO database. Nevertheless AUDIMUS was used to generate the alignment of the 5 h from the ALERT-SR.pilot set (the first available BN data). We used these 5 h of data to train our first BN acoustic model. This is referred as (Align 1) in the second line of Table 3.1. Its results show a huge improvement in WER. The next alignment (denoted Align 2 on Table 3.1) used 13 h of BN data for training the acoustic model (5 h from pilot set plus the first 8 h available from the training set) and again resulted in a performance increase. The alignment number 3 was performed when more training data became available (23 h). The language model was also updated using more newspaper text for estimating the 4-grams. Looking at the recognition results it might be seen that there was a decrease in performance. It can be explained by the increase in size of the ALERT-SR.devel test set (from 1 to 3 hours of data) so the WER results are not directly comparable. Nevertheless we almost doubled the size of the training set and observed only a negligible decrease in performance. With 13 h of training data and the MLPs topology with 500 hidden units we had in average 42 training samples per MLP weight. With 23 h this number increased to 75. According to [Ellis and Morgan, 1999] for this kind of task which has a lot of variability in the acoustic conditions the ideal number of training patterns per MLP weight should be between 20 and 40. In alignment number 4, denoted in Table 3.1 as Align 4a we increased the size of the MLPs in the acoustic model from 500 hidden units to 1000 hidden units. This change brought the average of training samples per MLP weight back to 37. Additionally we added an output class to represent speaker breath noises (inspirations and expirations). This is the only type of speaker noises present in the ALERT-SR corpus worth modelling. The other noise types do not have a significant number of occurrences. Nevertheless this had not a significant impact in performance. The changes in the acoustic model size and an additional realignment rendered an important gain in recognition performance. System Align 4b refers an update in the language model, the first one using interpolation of newspaper texts and BN transcriptions. Align 5a system featured a new realignment of training data, and an updated language model estimated in more newspaper texts. This was the first system evaluated in the full development test set data which has 6 hours. Align 5b was the first system using the new WFST dynamic decoder. We can see a small decrease in WER (from 33.6% to 32.0%). More important was the enormous decrease in decoding time from 30 xRT to 7 xRT (not shown in Table 3.1).

Alignment number 6 was the first to use the complete ALERT-SR.train set. From align 6 results we see that the WER reduction was insignificant (from 32% to 31.6%). To overcame this we increased again the size of the acoustic model MLPs. From 1000 hidden units to 2000 in align 7a and to 4000 hidden units in subsequent systems that used alignment number 7 (indicated in Table 3.1 as Align 7a through Align 7h). With 4000 hidden units the average of training samples per MLP weight is slightly less than 20. Combined also with two decoding improvements (determinization and minimization of the lexicon WFST and eager pruning [Caseiro and Trancoso, 2003]) the WER dropped significantly to 28.9% as can be seen for Align 7c results in Table 3.1.

Align 7d [Meinedo et al., 2003] reports one experiment with the minimum number of states in the phone HMMs. These HMMs have a minimum number of output states with a left to right topology, no transitions which skip over states and only a self-loop in the last state. The minimum number of phone states had been estimated using the BD-PUBLICO database and reflected the speaking rate of carefully read speech. This is not the most appropriate solution for BN data where there is a lot of spontaneous speech and where even prepared speech tends to have a much higher speaking rate. In this experiment we reduced the minimum number of states by one for all the phones that had a minimum number of states higher than 1. The recognition performance improved substantially (align 7d has 26.5% WER) although the decoding time doubles (not reported in Table 3.1). Align 7e shows the evaluation results of this same system in the ALERT-SR.eval test set which has 4 h of BN data. This test set has a higher WER when compared with the devel test set. Align 7f reports the results of an updated language model with more text training data and better discounting and pruning (Knesser-Ney discounting and entropy pruning as described in Section 3.3.2). This better LM yelded a significant reduction in WER. Align 7g and Align 7h show the evaluation results in ALERT-SR two other test sets: jeval and 11march respectively. This latter test set has a higher OOV rate and many sentences of spontaneous speech from live reports which are impaired by disfluencies degrading significantly the performance.

The last two lines of Table 3.1 (Align 8a and Align 8b) shows the evaluation for a new alignment having an updated acoustic model with a different MLP topology. This new acoustic model has MLPs with more input context (11 or 13 frames instead of 7) and has two hidden layers each one with 1000 or 1500 units. These changes motivated by [Zhu et al., 2005b] have induced a significant increase in performance. For Align 8b WER decreased 3% absolute from 26.7% to 23.7% in the ALERT-SR.eval test set. Earlier experiments with the former MLP topology and with a increased hidden layer from 4000 units to 8000 had showed us that simply increasing the number of parameters of the MLPs could not bring an improvement in performance. This is probably because we are reducing by half the number of training samples per network weight. In the same sense the new architecture has not showed yet its full potential since it is being trained with an average of less than 6 samples per network weight. It is expectable to have performance gains when the training set is augmented from the current 46 hours.

3.3.7 Gender dependent acoustic model

How to increase the recognition accuracy ? One solution is to use the ”Divide and conquer” approach to acoustic modelling and have several smaller models tuned for particular conditions (different speakers, different channel conditions (high bandwidth, telephone), different background conditions, etc). One of the most straight forward possibilities is to build gender dependent acoustic models, that is, have separate acoustic models for male and female. Of course, one needs to have a gender classifier to detect the sentence gender and use the appropriate acoustic model during recognition. In this work we have developed Audio Pre-Processing systems which incorporate gender classification and are described in detail in Chapter 4.

ALERT-SR.train set was divided into male and female sentences using the hand labeled classifications and gender dependent models were trained using Align 8c MLPs topology (13x26-1500-1500-40). The resulting system was evaluated in the ALERT-SR.jeval test set and compared against gender independent (gi) Align 8c. The results are present in Table 3.2. In the case of the gender dependent (gd) system (male + female) the sentences in the test set were recognized by the appropriate model (sentences with male speaker were recognized using the male acoustic model and sentences with female recognized with the female acoustic model). Overall performance decreased from 23.5% to 23.9%. Only the female gender dependent acoustic model represented slightly better performance. In the last line we used the gi acoustic model for recognizing male speech and the gender dependent female acoustic model for female speech. This lead to a small increase in performance.

Acoustic Model	% WER			
	F0	Male	Female	All
gi Align 8c	11.3	26.7	18.7	23.5
gd (male + female)	10.9	27.8	18.5	23.9
gi male + gd female	10.5	26.7	18.5	23.3

Table 3.2: AUDIMUS.media WER for gender dependent models in ALERT-SR.jeval test set.

3.4 AUDIMUS.media++ ASR system

Clearly we need more BN training data in order to build better acoustic models. The 46 hours of training data from ALERT-SR corpus were very important for developing a fully functional BN ASR system (AUDIMUS.media). Now to improve its performance we need to use a much larger training set in order to have an appropriate number of training examples for each phonetic class. For our current MLP architecture (13x26-1500-1500-40) the ALERT-SR.train set represents in average less than 6 training examples per weight. According to [Ellis and Morgan, 1999] for this kind of task the ideal number of training patterns per network weight should be between 20 and 40. This can help to explain why our gender dependent models did not improve much ASR performance (see previous Section). In one hand we are reducing data variability by having gender dependent models but in the other hand we are also reducing dramatically the number of training patterns for each phonetic classifier.

The solution for this problem is to have a much larger BN speech data training set. But the process of collecting and annotating speech data is very expensive both in terms of time and money. So the solution for the problem is to automatically collect and annotate the BN data. But automatically annotated data has mistakes so it is necessary to have an unsupervised selection process using confidence measures to choose the most accurately annotated speech portions and add them to the training set.

According to [Wessel and Ney, 2005] it is possible to start with a relatively small hand labelled speech data training set and to add automatically more training data using confidence measures. It is known in literature that this process can lead to better ASR performance. Following these approaches we started to reuse automatically collected data for training. The data collection part was carried by our Media Monitoring prototype which is described in Chapter 5. The collected speech data was called SSNT-BN corpus and is described in detail in Chapter 2. The first experiment consisted of using one complete month of RTP station 8 o'clock news shows ("Telejornal"). This data set was called SSNT-BN.one-month-2005. After the success of the first experiment, the second experiment was to add all available editions of RTP2 main news show which starts daily at 10 o'clock in the night ("Jornal2"). This data set is called SSNT-BN.jornal2. The third experiment consisted in adding more six months of "Telejornal" news shows from 2006. This data set is called SSNT-BN.six-months-2006.

The phonetic classification and recognition results for AUDIMUS.media++ were obtained in the ALERT-SR.jeval test set. Table 3.3 shows WER results for the new acoustic models. As we can see the addition of more training data always lead to better recognition performance.

Training set	Amount of data	% WER	
		F0	All
ALERT-SR.train (Align 8c)	46 h	11.3	23.5
" + one-month-2005	63 h	11.2	23.2
" + jornal2	79 h	11.0	22.7
" + six-months-2006	179 h	10.8	22.1

Table 3.3: AUDIMUS.media++ WER in ALERT-SR.jeval test set.

3.4.1 Speaker adapted acoustic models

In the news shows being processed by our media monitoring system the news anchors are responsible for producing a significant amount of speech (around 25% of the total news show duration). This alone justifies building adapted acoustic models for these very frequent speakers, that is, if they are not replaced too often (which is the case). Looking at the WER results obtained by our best acoustic model in the ALERT-SR.jeval test set (reproduced in the first line of results of Figure 3.4) we see that Anchor A ("José Rodrigues dos Santos") has a WER of almost 20%.

As a first of building an adapted acoustic model for this news anchor we started from Align 8c acoustic MLPs and adapted them to all of this speaker speech available in ALERT-SR.train set. This represents around 2 hours of useful speech. The results of this adaptation are present in Table 3.4.

Acoustic Model	% WER	
	Anchor A	All
Align 8c	19.4	23.5
Align 8c + SD for Anchor A	17.7	23.2

Table 3.4: AUDIMUS.media++ WER for speaker dependent models in ALERT-SR.jeval test set.

There was already a gain in recognition performance by using an adapted acoustic model even when the same data had been used for training the speaker independent acoustic model. This opens good perspectives for building more complex BN acoustic dependent models.

3.5 Summary

This chapter described the modifications that transformed our European Portuguese ASR system developed for a dictation task into a system suitable for processing BN speech. Our ASR system evolved substantially with the availability of large quantities of BN training data. This new data permitted the development of acoustic and language models appropriate for the BN speech recognition task.

Our current BN ASR system had a good performance, although the results for European Portuguese are not yet at the level of the ones for languages like English, where much larger amounts of training data are available. We believe that unsupervised training approach described will be very helpful in this context and will permit us to continue to lower the WER and to use other approaches for enhancing the recognition besides gender dependent models like context dependent models or speaker adaptations. Nevertheless although the recognition is not yet perfect (22.1% WER) our ASR system is able to drive successfully a media monitoring system.

Chapter 4

Audio Pre-Processing

Broadcast News media monitoring is an important technology but poses a number of difficulties and challenges for speech processing both in terms of computational complexity and transcription accuracy. In this kind of application the speech signal not only has to be transcribed but also characterized in terms of acoustic content. Most present day transcription systems perform some kind of audio characterization (segmentation and labelling) as a first step in the processing chain [SPECOM, 2002]. To accomplish this audio characterization the first stage in our media monitoring system is an Audio Pre-Processing (APP) module. It is responsible for $i)$ the segmentation of the signal into acoustically homogeneous regions, $ii)$ for classification of those segments according to speech and non-speech intervals, background conditions, speaker gender and $iii)$ for identifying all segments uttered by the same speaker. The segmentation provides information regarding speaker turns and identities allowing for automatic retrieval of all occurrences of a particular speaker. The segmentation can also be used to improve performance through adaptation of the speech recognition acoustic models. Additionally the final transcriptions enriched by the pre-processing information are somewhat more human readable. APP offers some practical advantages: no waste of time on the processing of non-speech intervals, no need to process very long speech chunks, facilitation of gender or speaker dependent acoustic model selection during recognition. On the other segmentation errors may cause extra transcription errors if an acoustic change is hypothesized in the middle of an utterance or even worse in the middle of a word (In Chapter 5 we present the impact in terms of recognition errors of using automatic Audio Pre-Processing).

This chapter describes in detail the development of our APP module. It starts by explaining the global (canonical) architecture of an APP module and by giving examples of what the components perform. Then it gives a detailed description of the methodologies used in the training and evaluation. The following sections describe our APP modules developed during this thesis work: version 1.0, version 2.0 and version 3.0. We end the chapter with a short summary that reviews the work done.

4.1 Canonical architecture

Our APP module and all the APP algorithms described in literature share a more or less similar canonical architecture whose block diagram is represented in Figure 4.1. This canonical architecture is composed by an audio input signal fed to a chain of processing blocks that at the end output a series of characterizations describing the audio. The processing blocks are normally divided into 3 categories: audio segmentation, audio classification and speaker classification.

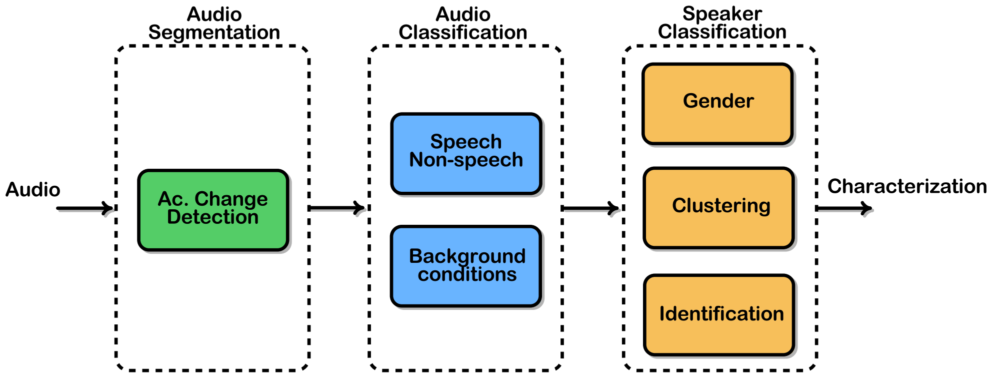

Figure 4.1: Audio Pre-Processing canonical block diagram

The audio segmentation usually has only one processing block, called Acoustic Change

Detection (ACD) which is responsible for the detection of audio locations where speakers or background conditions have changed. The audio classification category has 2 processing blocks, one for Speech/Non-speech (SNS) classification which is responsible for classifying the segments in order to verify if they contain useful speech or not, the other block is for Background Conditions (BC) classification that detect whether the background is clean has noise or music. The speaker classification category has 3 processing blocks, one for Gender Detection (GD) that distinguish between male and female gender speakers, one for Speaker Clustering (SC) which groups together all the segments produced by the same speaker and one processing block for Speaker Identification (SID) which is responsible for identifying certain relevant speakers, like anchor speakers in the news show. Their identification will be useful for Topic Segmentation (TS) and Topic Indexing (TI) since normally a new story is introduced by the anchor person.

Figure 4.2 represents an example of the audio characterization done by our APP module. First the audio is segmented into homogeneous regions by the Acoustic Change Detection (ACD) block. All segments are classified according to Speech/Non-speech (SNS) and Background Conditions (BC). The speech segments are classified according to the gender by the Gender Detection (GD) block and segments uttered by the same speaker are marked by the Speaker Clustering (SC) block using a unique cluster number. Finally, the Speaker Identification (SID) block detects if the segments belong to one of the known speakers.

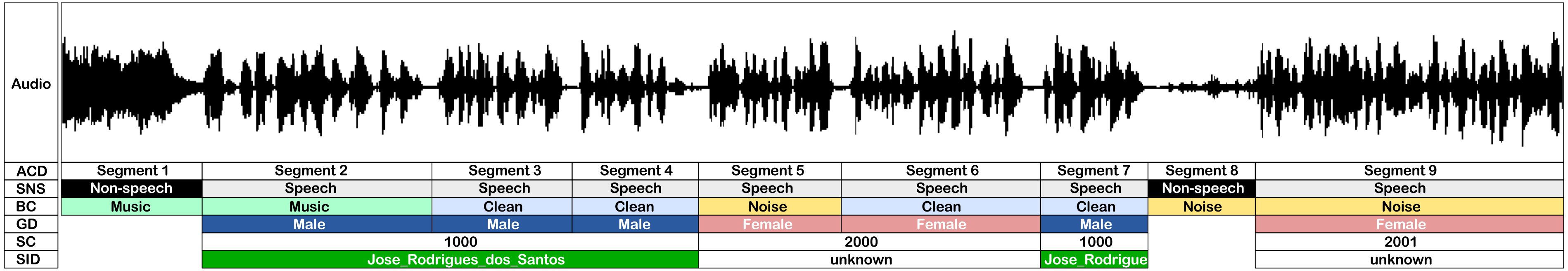

Figure 4.2: Audio Pre-Processing example

4.2 Training and evaluation methodologies

For the training and evaluation of the APP components two different databases were used, the ALERT-SR and the COST278-BN. The ALERT-SR train and pilot sets were used to train all our APP components. The ALERT-SR.devel was used for tuning parameters such as thresholds. For the evaluation two different test sets were used. The first one was the ALERT-SR.jeval because it is representative of the news shows our media monitoring system has to process. If the evaluation shows a better performance it is likely that our media monitoring system will also have better performance. The other reason for using the jeval test set is because it permits speaker identification evaluations since it has the speakers found in our SID blocks. The second test set used was the COST278-BN corpus. Using this database we were able to compare our APP modules with other algorithms specially those that were developed by our partners in the European COST278 action [Vandecatseye and Martens, 2003, Zdansky et al., 2004]. Additionally the evaluation tools used were also developed within the COST278 action and were common to all partners.

4.2.1 Evaluation measures and tools

To evaluate our APP algorithms it became necessary to develop an appropriate set of tools since the existing ones where unavailable or inadequate. NIST has a simple script for evaluating ACD [NIST, 2000] but it has limitations and it proved to be inadequate. NIST also developed an evaluation script which computes the Diarization Error Rate (DER) is the “standard” measure used for evaluating speaker clustering. This is the only measure used for evaluating APP modules in NIST Rich Text (RT) evaluations [NIST, 2004]. For the other components of the APP we found no appropriate tools for evaluating their performance.

We developed independent evaluation tools for evaluating all the blocks that constitute our APP modules. Our tool for evaluating ACD performance uses standard performance measures: Recall, Precision and F-measure. Recall can be defined as the percentage of detected acoustic change points. Precision is the percentage of detected points which are genuine change points and F-measure takes into account both Precision and Recall. In order to calculate these figures the evaluation software performs a one-to-one mapping between computed and reference segmentation points given a maximum tolerance of 1 second around each reference time boundary.

$$\text{Recall} = \frac{\text{correct change points found}}{\text{total number of change points}} \tag{4.1}$$

$$\text{Precision} = \frac{\text{correct change points found}}{\text{total number of change points found}} \tag{4.2}$$

$$\text{F-measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.3}$$

Reference change points refer only to speaker changes and not background condition changes. Speech/Non-speech (SNS), Background Conditions (BC) and Gender Detection (GD) classification results are reported in terms Classification Error Rate (CER), defined as the ratio between the number of incorrectly classified frames and the total number of frames. Additionally the evaluation tools report the percentage of incorrectly classified frames for each class [Speech, Non-speech] for the SNS case, [Clean, Music, Noise] for the BC case and [Male, Female] for the GD case.

$$\text{CER} = \frac{\text{incorrectly classified frames}}{\text{total number of frames}} \tag{4.4}$$

In order to evaluate the Speaker Clustering and the Speaker Identification a bi-directional one-to-one mapping of reference speakers to clusters was computed (NIST rich text transcription evaluation script). The most often used performance measure for clustering is the overall Diarization Error Rate (DER) which is computed as the percentage of frames with an incorrect cluster-speaker correspondence.

$$\text{DER} = \frac{\text{frames with incorrect cluster-speaker correspondence}}{\text{total number of frames}} \tag{4.5}$$

Other performance measures used were the average Purity Error Rate (PER) for clusters and for speakers. Cluster PER is defined as the ratio between the number of frames not belonging to the dominant speaker in a cluster and the total number of frames in the cluster. Speaker PER accounts for the dispersion of a given speakers data across clusters. It is defined as the ration between the number of frames not belonging to the dominant cluster in the speaker and the total number of frames of the speaker.

$$\text{average clusters PER} = \sum_k \frac{\text{frames not from dominant speaker of cluster } k}{\text{total number of frames of cluster } k} \tag{4.6}$$

$$\text{average speakers PER} = \sum_k \frac{\text{frames not from dominant cluster of speaker } k}{\text{total number of frames of speaker } k} \tag{4.7}$$

No cluster information was passed between different files.

4.2.2 Comparisons with other algorithms

Within the COST278 action our APP module was compared with algorithms developed by partners from seven research institutions: ELIS (Gent), TUB (Budapest), TUK (Kosice), TUL (Liberec), ULJ (Ljubljana), UMB (Maribor) and UVIGO (Vigo). This had the advantage of simplifying direct comparisons without the need for implementing all state of the art algorithms because we used the same evaluation data, protocols and tools.

4.3 Audio Pre-Processor version 1.0

The first version of our Audio Pre-Processing module was developed for aiding the annotation process of ALERT-SR corpus. We needed a working system even if its performance was not the best. Together with the initial version of AUDIMUS.media described in Chapter 3 it composed the core of our first BN automatic transcription system that helped enormously to speed up the BN corpus annotation. This first APP module worked offline and without real time constraints although we will show that it only takes a fraction of real time to process the audio. The architecture is shown in Figure 4.3. As we can see its block diagram is very similar to the general purpose one presented in Figure 4.1. The purpose of the acoustic change detection module is to generate homogeneous acoustic audio segments. The segmentation algorithm detects changes in the acoustic conditions and marks those time instants as segment boundaries. Each homogeneous audio segment is then passed to the first classification stage in order to tag non-speech segments. All audio segments go through the second classification stage where they are classified according to background conditions. Segments that were marked as containing speech are also classified according to gender. All labelled speech segments are clustered separately by gender in order to produce homogeneous clusters. The speaker identification is performed after the speaker clustering. This SID block tags speaker clusters that belong to one of the pre-defined news anchors.

4.3.1 Acoustic Change Detection

The main goal for the ACD is to divide the input audio stream into acoustically homogeneous segments. This is accomplished by evaluating in the cepstral domain the similarity

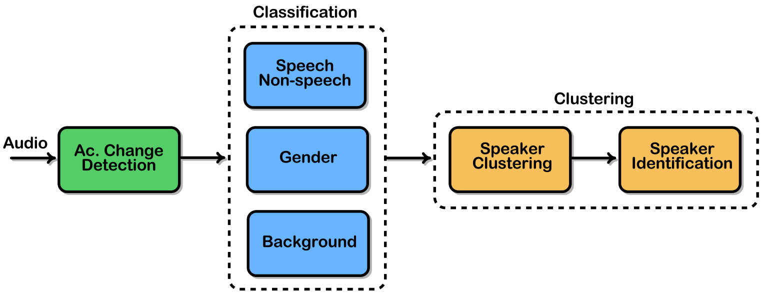

Figure 4.3: APP version 1.0 block diagram.

between two contiguous windows of fixed length that are shifted in time every 10 ms. We used the symmetric Kullback-Liebler, KL2 [Siegler et al., 1997], as the distance measure to evaluate acoustic similarity. Each window is modelled by a Gaussian distribution. Large values for the KL2 imply that the distributions of the windows are more dissimilar. The KL2 is calculated over 12^{th} order PLP [Hermansky et al., 1992] coefficients extracted from the audio signal. We considered a segment boundary when the KL2 distance reached a maximum. The maxima values are selected using a pre-determined threshold detector. The diagram of our ACD block is shown in Figure 4.4.

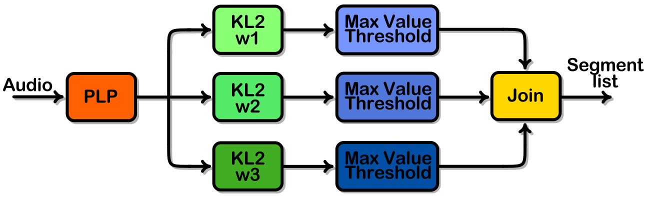

Figure 4.4: KL2 ACD.

This ACD block uses three distinct time analysis window pairs of 0.5, 1.0 and 4.0 seconds [Meinedo and Neto, 2003a]. Small analysis windows can obtain a higher degree of time accuracy. Larger windows have less time accuracy but are able to detect slower audio transitions. The final segment transition list is a weighted sum of the three transition lists evaluated inside a 50 msec window. More weight is given to the more time accurate window pairs. Table 4.1 highlights the results obtained in the COST278-BN corpus by the ACD with the standard KL2 single window pair of 0.5 sec (the size which obtained best results) and the improved scheme with three different KL2 window sizes.

Using the improved scheme of analysis windows with different sizes the number of missed

ACD	Recall	Precision	F-measure
single KL2, 0.5 sec	46.0	55.7	50.3
three KL2, 0.5, 1.0, 4.0 sec	71.8	59.5	64.5

Table 4.1: ACD evaluation in COST278-BN database.

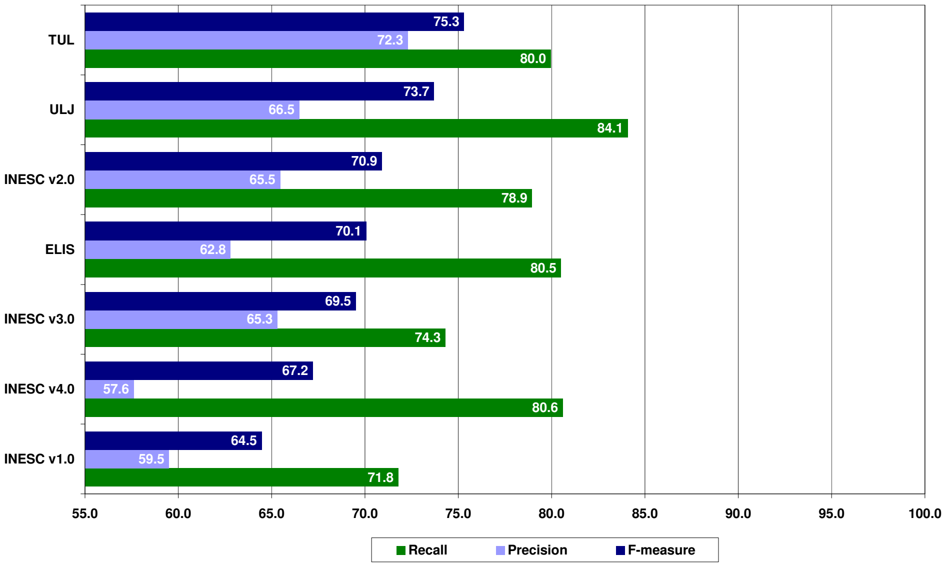

Figure 4.5: COST278-BN comparison results for ACD (systems are sorted by F-measure).

boundaries was reduced significantly (thus having a higher Recall). Overall result translated by F-measure show good improvement. Comparing our ACD block with other algorithms developed by some of our COST278 partners [Zibert et al., 2005] we see in Figure 4.5 that our algorithm obtained the worst F-measure. It can be partially explained by a high sensitivity to background conditions which cause a high rate of insertions and the effect of using diagonal covariance matrices versus full covariance ones. Additionally we observed in the SI2 data, which had a known audio amplitude deficiency, that the F-measure value obtained is 10% absolute below all other language sets (Probably a lack of robustness of this segmentation method to audio data with insufficient amplitude).

4.3.2 Speech/Non-speech classification

After the ACD stage each segment is classified using a Speech/Non-speech (SNS) discriminator. Ideally a segment is tagged as “Non-speech” when it contains no speech or the speech is not “understandable” because of degraded conditions like too much noise. This stage is very important for the rest of the processing in order not to waste time trying to recognize segments that do not contain “useful” speech.

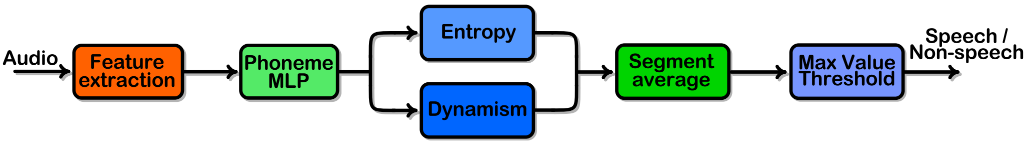

Figure 4.6: SNS classification using entropy and dynamism features.

Figure 4.6 represents the SNS classification block. 12^{th} order PLP coefficients are extracted from the audio signal. These feature vectors are input into a Multi-Layer Perceptron (MLP) that was trained to estimate context-independent phone posterior probabilities. This MLP is the same used as acoustic model by our hybrid HMM/MLP recognition system AUDIMUS.media. It was trained using 23 hours of BN data from the ALERT-SR corpus. It has an architecture with 7 context input frames of 26 features (12^{th} order PLP coefficients plus energy and delta features) a hidden layer with 1000 sigmoidal units and 40 softmax output units representing the 38 phones of the European Portuguese plus silence and breath noises. Local posterior probabilities estimated by the MLP are used to calculate two acoustic confidence measures: instantaneous per-frame entropy and the probability dynamism [Williams and Ellis, 1999]. The entropy of the K posterior probability estimates associated with HMM states q_k is defined as,

$$H(n) = -\sum_{k=1}^{K} P(q_k|x^n)\log(P(q_k|x^n)) \tag{4.8}$$

where x_n is the acoustic vector at time n and $P(q_k|x^n)$ the posterior probability of phone q_k given x_n at the input. Low values for the entropy indicate regions where the acoustic model provides a good match to the observed input data, since the distribution of phone posteriors will be dominated by a single class phone. High values of entropy represent more uniformly distributed probability values and indicate regions of poorly modelled audio by the acoustic model and are likely candidates to be regions of music, noise or very degraded speech. This instantaneous entropy measure is inherently noisy due to phone transitions during normal speech and a median filter with a 0.5 sec window was used to smooth the output. Finally the average value for the segment is calculated. The probability dynamism measures the rate of change in phone probability estimates and is given by

$$D(n) = \sum_{k=1}^{K} (P(q_k|x^{n-1}) - P(q_k|x^n))^2 \tag{4.9}$$

The value for dynamism in normal speech is high because probability estimates for well modelled speech segments change abruptly and frequently. Non-speech signal are typically less varying and consequently will receive lower dynamism values. Again, the average for one audio segment is calculated. Both acoustic confidence measures are thresholded and used to take a decision whether the has speech or non-speech.

	% Speech	% Non-Speech	CER
Speech/Non-speech	3.0	29.8	4.9

Table 4.2: SNS evaluation in COST278-BN database.

Table 4.2 represents the evaluation results in the COST278-BN database obtained by the SNS block. This classifier has a low overall CER and a very low error rate of 3.0% for tagging speech segments as non-speech. It is the worst error since these segments had useful speech and will not be sent to the ASR. In that sense we can say that our algorithm has good results.

Comparing this SNS classification module with the algorithms developed by our partners in the COST278 [Zibert et al., 2005] we see in Figure 4.7 that our algorithm (INESC v1.0) ranked in 7th place out of 10. This somewhat low classification pushed us to develop better SNS algorithms.

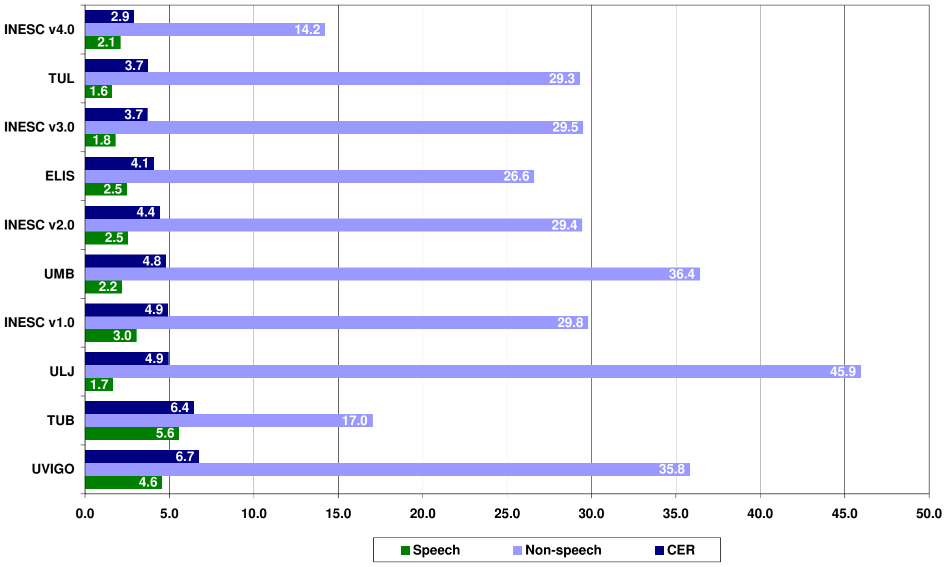

Figure 4.7: COST278 comparison results for SNS classification (systems are sorted by CER).

4.3.3 Gender Detection

In our framework Gender Detection (GD) is used as a mean to improve speaker clustering. By separately clustering each gender class we will have a smaller distance matrix when evaluating cluster distances which effectively reduces the search space. It also avoids short segments having opposite gender tags being erroneously clustered together.

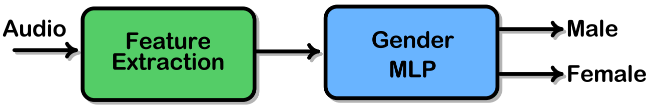

Figure 4.8: Gender Detection module

The GD module shown in Figure 4.8 uses one MLP estimating gender posterior probabilities. This classifier has 9 input context frames of 12^th order PLP features and a hidden layer with 250 sigmoidal units. The gender MLP has two output classes, male and female. The output of the module is obtained by calculating the class that has the highest average probability over all the frames of the segment. The classifier was trained using the ALERT-SR.train set. Table 4.3 summarizes the results obtained by the GD module evaluated in the COST278-BN corpus. The evaluation result represented in Table 4.3 show that this GD module exhibits low classification error rates.

	% Male	% Female	CER
Gender	3.8	10.2	6.0

Table 4.3: GD evaluation in the COST278-BN database.

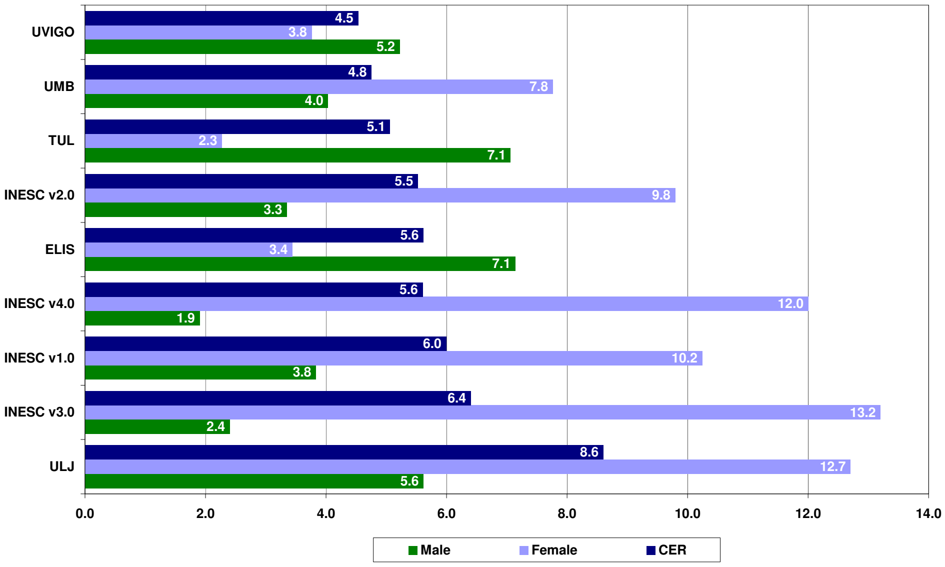

Figure 4.9: COST278 comparison results for GD (systems are sorted by CER).

Figure 4.9 shows the COST278 comparative evaluation. When we compare our GD module with other algorithms developed by our partners in the COST278 [Zibert et al., 2005] we see in Figure 4.9 that our algorithm obtains again very good results ranking amongst the best ones.

4.3.4 Background Conditions Classification

Background Conditions (BC) classification helps to characterize the audio signal and can be used in TS modules to detect story segments. The BC classification module shown in Figure 4.10 uses one MLP estimating posterior probabilities. It uses an MLP with 9 input context frames of 12^th order PLP features and a hidden layer with 250 sigmoidal units. It has three output classes, clean, noise and music. The output of the module is obtained by calculating the class that has the highest average probability over all the frames of the segment. The classifier was trained using the ALERT-SR.train set.

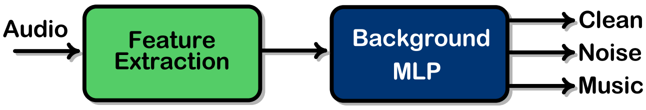

Figure 4.10: Background classification module.

Table 4.4 shows the results obtained when testing the BC module in the COST278-BN corpus. This BC module exhibits a moderate CER. We found that the BC module has a difficult task because in our training material there are many overlapping, especially music plus noise. Furthermore we found that many hand annotated segments have dubious classifications when certain noises corrupt a normal clean background (is it clean or noise ?).

	% Clean	% Music	% Noise	CER
Background	34.5	41.5	46.8	40.2

Table 4.4: BC evaluation in the COST278-BN database.

4.3.5 Speaker Clustering

The goal of Speaker Clustering (SC) is to identify and group together all speech segments that were produced by the same speaker. The clusters can then be used for an acoustic model adaptation in order to improve the speech recognition rate. Speaker cluster information can also be used by topic detection and story segmentation algorithms to determine speaker roles inside the news show allowing for easier story identification. Our speaker clustering algorithm uses GD information. Speech segments with different gender classification are clustered separately. We used bottom-up hierarchical clustering [Siegler et al., 1997]. In this approach speech segments are modelled in the cepstral domain by Gaussian distributions. Initially each segment is considered a cluster. The algorithm computes a distance matrix for all clusters and the two closer ones are considered for joining in a new cluster. Clusters are linked together until the distances exceed a pre-defined value. At that point the clustering ends. Several appropriate distance measures can be used, namely the KL2 [Siegler et al., 1997], the generalized likelihood ratio or the BIC [Chen and Gopalakrishnan, 1998][Zhou and Hansen, 2000]. Our first experiments were conducted using the KL2 metrics to evaluate cluster distances. Latter on we developed a more efficient distance measure based on the BIC. The distance measure when comparing two clusters using the BIC can be stated as a model selection criterion where one model is represented by two separated clusters C_1 and C_2 and the other model represents the clusters joined together $C = \{C1, C2\}$. The BIC expression is given by,

$$BIC = nlog|\Sigma| - n_1log|\Sigma_1| - n_2log|\Sigma_2| - \lambda P \tag{4.10}$$

where $n = n_1 + n_2$ gives the data size, Σ is the covariance matrix, P is a penalty factor related with the number of parameters in the model and λ is a penalty weight. If $BIC < 0$ the two clusters are joined together.

We made two modifications to this criterion. First, we considered that the Gaussian distributions had diagonal covariance matrices, that is, we considered that the features were uncorrelated. Our speaker clustering tests showed that this modified BIC performs better than the KL2 and at the same time is much less computationally intensive than the full BIC. Second, an adjacency term is used instead of the BIC threshold λ. The new penalty weight is now given by $\lambda = 1 + \alpha k$, where k represents the number of adjacent speech segments between both clusters C_1 and C_2 and α is a positive constant. If the clusters do not have adjacent segments, $k = 0$ and consequently $\lambda = 1$. When they have adjacent segments, $\lambda > 1$, and λP will be larger favouring the model where both clusters belong together. Empirically clusters having adjacent speech segments are closer in time and the probability of belonging to the same speaker must be higher. Table 4.5 illustrates the results obtained by our SC module in the COST278-BN corpus.

Clustering	PER clus	PER spkr	DER
KL2	25.5	38.2	40.7
modified-BIC	18.6	35.4	37.3
modified-BIC + adjacency	17.6	34.5	35.6

Table 4.5: SC evaluation in the COST278-BN database.

Normally, a higher cluster purity is more desirable and less costly for subsequent processing than a smaller number of clusters per speaker. Looking at the results of Table 4.5 we see that the adjacency term in the modified BIC expression retained a high cluster purity and decreased significantly the number of clusters per speaker (lower DER). The clustering algorithm proved to be sensitive not only to different speakers but also to different acoustic background conditions. This side-effect is responsible for the high number of clusters per speaker obtained in the test set results (higher DER values).

Figure 4.11 shows the DERs for the different SC modules that participated in the join COST278 evaluation [Zibert et al., 2005].

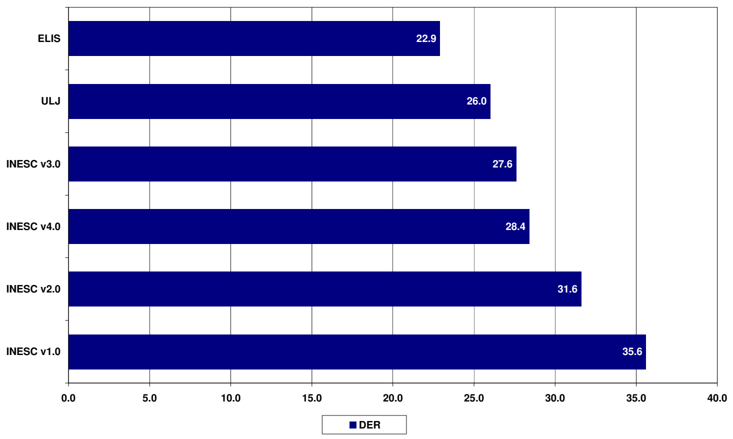

Figure 4.11: COST278 comparison results for SC (systems are sorted by DER).

Our SC module has the largest DER mainly because it generates more clusters. Since all systems generate more clusters than speakers more clusters means more incorrectly classified clusters.

4.3.6 Speaker Identification

Anchors introduce the news and provide a synthetic summary for the story. Normally this is done in studio conditions (clean background) and with the anchor reading the news. Anchor speech segments convey all the story cues and are invaluable for automatic topic and summary generation algorithms. Also in these speech segments the recognition error rate is the lowest possible. The news shows that our media monitoring system is currently processing are presented by three anchor persons, two male and one female. As first approach to a Speaker Identification (SID) module we built individual speaker models for these three anchors persons. Each model is composed of sentence clusters representing speech from the anchor in different background conditions. Typically a model does not have more than nine clusters. Each anchor model was built using sentences from 2 news shows of over 1 hour. During the processing of a news show after SC the resulting clusters are compared one by one against the special anchor cluster models to determine which of those belongs to one of the news anchors. This cluster comparison uses the KL2 distance metrics to measure cluster similarity. If the KL2 value is lower than a specified threshold the cluster is tagged as an anchor cluster.

Clustering	PER clus	PER spkr	DER
+ speaker models	17.5	34.2	35.6

Table 4.6: SID evaluation in the COST278-BN database.

Table 4.6 shows the results obtained in the COST278-BN database. Since these anchor persons are only present in the Portuguese data of the COST278-BN the improvement is negligible.

Anchor detection evaluation presents some difficulties. Anchor persons tend to change more frequently than desirable and the identification does not work for the other languages of the COST278-BN database.

4.4 Audio Pre-Processor version 2.0

From the evaluations done in cooperation with our partners at the COST278 action it was clear that our APP system had much room for improvement. Also a near-term objective for our media monitoring system is to be able to function in real-time in order to provide automatic close-captioning. To accomplish this not only the APP and ASR modules must have been designed for online processing (capable of working in stream-based mode) but also the system needs to have faster than real-time processing time. It is also a constraint that this must be achieved with low constant latency so not to broaden the gap between input signal and output transcriptions. Furthermore, in an attempt to model more precisely the audio signal without increasing significantly the complexity, we changed some modules to extensively use Artificial Neural Network (ANN) models. This new version (named 2.0) of our APP has low latency and is stream-based. It performs Speech/Non-speech (SNS) classification, Acoustic Change Detection (ACD), Gender Detection (GD), Background Conditions (BC) classification and Speaker Clustering (SC). Again evaluation experiments were conducted on the COST278-BN multi-lingual TV broadcast news database and compared with the algorithms developed by our COST278 partners using the standard evaluation tools. Additionally we tested the performance of all modules in ALERT-SR.jeval test set because it is representative of the news shows our media monitoring system has to process.

System version 2.0 shown in Figure 4.12 is composed by five modules: three for classification (Speech / Non-speech, Gender and Background), one for speaker clustering and one for acoustic change detection. All five modules are model-based meaning that they incorporate algorithms trained using a priori information (from databases). As a way to increase the modelling accuracy our algorithms make extensive use of Artificial Neural Networks (ANN) thus avoiding the rough assumptions normally made about the audio signal distribution, namely the assumption that fairly large blocks of audio (2 to 3 seconds) can be approximated by Gaussian distributions. All ANN used are of the type feed-forward fully connected Multi-Layer Perceptron (MLP) and were trained with back-propagation algorithm.

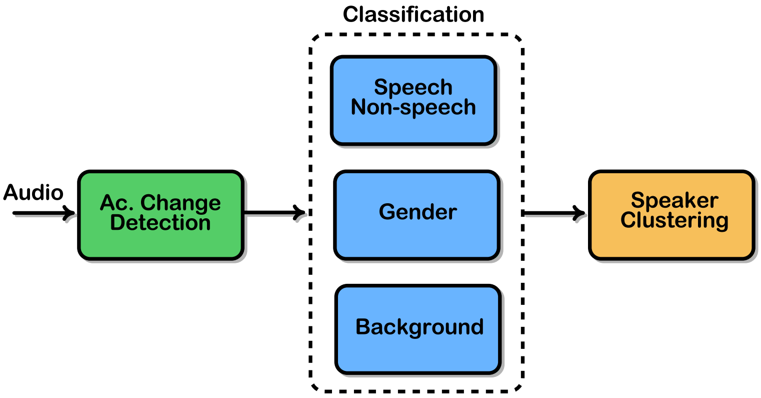

Figure 4.12: Audio pre-processing system version 2.0 overview.

4.4.1 Speech/Non-speech Classification

The Speech/Non-speech module is responsible for identifying audio portions that do not contain speech, with too much noise or pure music. This serves two purposes: first, no time will be wasted trying to recognize audio portions that do not contain speech; second, reduces the probability for speaker clustering mistakes. The architecture represented in Figure 4.13 is composed by a MLP with 9 input context frames of acoustic features each one with 26 coefficients. That is, 12^{th} order PLP features plus log energy plus deltas. The MLP has an hidden layer with 300 sigmoidal units. And two complementary outputs for estimating the probability of a given input frame containing speech or non-speech. When the Acoustic Change Detector hypothesized the start of a new segment, the first N_{class} frames of that segment are used to calculate the speech/non-speech, gender and background classification. The output of the module is obtained by calculating the class that has the highest average probability over all the frames of the segment. After initial experiments in the training set, N_{class} was set to 300 frames. This relatively short interval is a trade-off between performance and the desire for a very low latency time. The MLP classifiers for SNS, GD and for BC were trained using the ALERT-SR.train set.

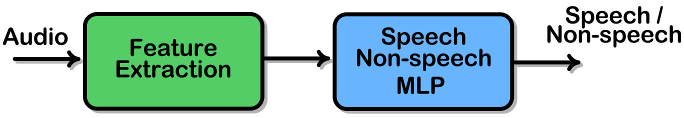

Figure 4.13: SNS module version 2.0.

	% Speech	% Non-speech	% CER
Speech/Non-speech	2.5	29.4	4.4

Table 4.7: SNS evaluation in COST278-BN database.

Looking at the results represented in Table 4.7 for the COST278-BN database we see that this new version has improved performance when compared with version 1.0 which already had excellent results. This SNS classifier MLP is substantially smaller that version 1.0. It can now be trained using databases without phonetic information (like a non annotated foreign language database) while version 1.0 required labelled phonetic annotation to train the phonetic classifier.

4.4.2 Gender Detection

In version 2.0 Gender Detection is still used to improve speaker clustering. By clustering separately each gender class we have a smaller distance matrix when evaluating cluster distances which effectively reduces the search space. It also avoids short segments having opposite gender tags being erroneously clustered together. Figure 4.14 is composed by a MLP with 9 input context frames of acoustic features each one with 26 coefficients. That is, 12^{th} order PLP features plus log energy plus deltas. The MLP has an hidden layer with 300 sigmoidal units. And two complementary outputs for estimating the probability of a given input frame containing male or female speech.

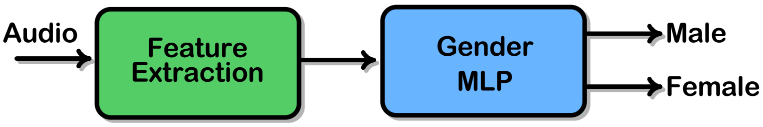

Figure 4.14: GD module version 2.0.

Looking at the results represented in Table 4.8 for the COST278-BN database we see that

	% Male	% Female	% CER
Gender	3.3	9.8	5.5

Table 4.8: GD evaluation in COST278-BN database.

this new version has slightly improved performance when compared with version 1.0.

4.4.3 Background Conditions classification

Currently Background classification is being used to enrich the metadata in the final transcription XML file and to aid Topic Detection. The BC module has the same architecture of version 1.0. It is composed of a MLP with 9 input context frames of acoustic features each one with 26 coefficients. That is, 12^{th} order PLP features plus log energy plus deltas. The MLP has an hidden layer with 300 sigmoidal units which is a slightly increase in size when compared with version 1.0.

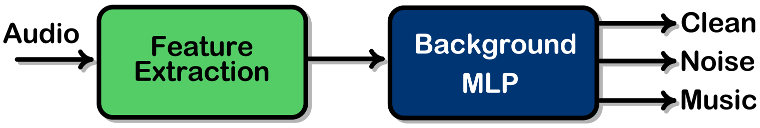

Figure 4.15: BC module version 2.0.

	% Clean	% Music	% Noise	CER
Background	30.2	41.1	45.3	39.8

Table 4.9: BC evaluation in COST278-BN database.

Evaluation was again conducted using the COST278-BN database [Vandecatseye et al., 2004] and the standard tools developed by the COST278-BN SIG. Table 4.9 summarizes BC results in the COST278-BN database. Background classification performance increased in version 2.0 although only slightly.

4.4.4 Acoustic Change Detector

The main goal for the Acoustic Change Detector is to divide the input audio stream into acoustically homogeneous segments. This module uses a hybrid two stage algorithm combining energy, metric and model based techniques and is represented in Figure 4.16. During the first stage a large set of candidate change points are generated. In the second stage these candidate change points are evaluated again and some that do not correspond to true speaker change boundaries are eliminated.

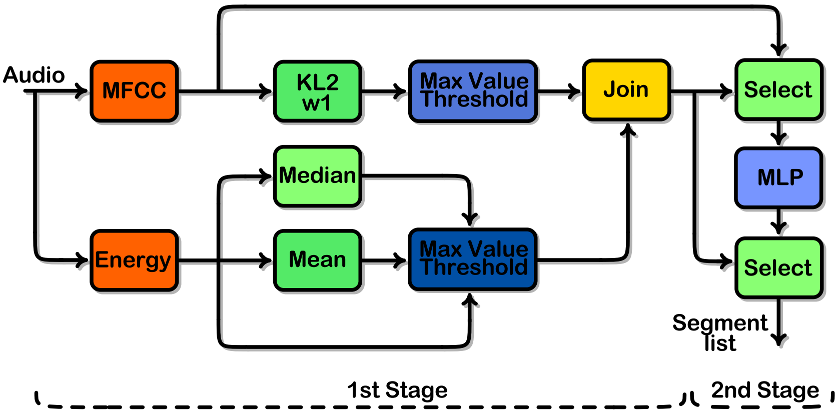

Figure 4.16: Acoustic Change Detection version 2.0.

The first stage uses two algorithms to generate the set of candidate boundaries. The first one is metric-based. This is accomplished by evaluating, in the feature cepstral domain, the similarity between two contiguous windows of fixed length that are shifted in time every 10 ms. We used the symmetric Kullback-Liebler, KL2 [Siegler et al., 1997], as the distance measure to evaluate acoustic similarity. The KL2 is calculated over 12^{th} order PLP coefficients extracted from the audio signal. We considered a segment boundary when the KL2 distance reached a maximum. The maxima values are selected using a pre-determined threshold detector. The second algorithm is energy-based. Instantaneous energy is calculated in a frame basis. From this energy values two measures are derived, the median filtered by a 50 frame window and a long term average of 250 frames. A threshold detects when the median signal drops bellow the long term average (small and big pauses in speech discourse). Of course these pauses may not correspond to speaker change (our main goal) but generally they do. These two algorithms complement themselves: energy is good on slow transitions (fade in/out) where KL2 is limited because of fixed length window. Energy tends to miss the detection of rapid speaker changes for situations with similar energy levels while KL2 does not. The second stage uses a MLP classifier to make a decision whether the candidate boundary should be removed or not. The input of the classifier uses a huge 300 frames context ($N_{acd} = 3$ sec) of acoustic features with 12^{th} order PLP plus log energy and a hidden layer with 150 sigmoidal units.

The MLP classifier was trained by generating candidate boundaries for the whole ALERT-SR.train set using the first stage and then aligning those boundaries using the reference boundaries. This procedure generated an appropriate boundary training set with balanced positive and negative examples. By using audio feature vectors directly into the MLP input we are using a more precise signal model and not assuming that the *PDF* of the features of the audio signal is Gaussian in the N_{acd} window.

The ACD module is responsible for triggering SNS, GD and BC classifiers when detects that a new segment is beginning. A new segment is only hypothesized if the non-speech portion is greater that 1.5 seconds or if the clusters do not match.

	Recall	Precision	F-measure
Ac. Change Detector	78.9	65.5	70.9

Table 4.10: ACD results in COST278-BN database.

Table 4.10 represent the ACD results for the evaluation in the COST278-BN database. We can see an improvement of 6.4% compared with version 1.0, from 64.5% to 70.9%. When compared with our COST278 partners (Figure 4.5) we are better placed.

4.4.5 Speaker Clustering

The goal of speaker clustering is to identify and group together all speech segments that were uttered by the same speaker. Our algorithm works in the following way: after the acoustic change detector signals the existence of a new boundary and the classification modules determine that the new segment contains speech of male/female gender, the first N_{clus} frames of the segment are compared with all clusters found so far. The segment is merged with the cluster for which the lower distance was calculated if bellow a predefined threshold. The distance of a segment to a cluster is given once more in a model-based approach by a MLP. This classifier was trained to estimate the probability of a given segment of acoustic features belonging to a particular cluster also specified in terms of audio feature vectors. Our speaker clustering algorithm makes use of gender detection. Speech segments with different gender classification are clustered separately. There are two MLP classifiers, one for each gender type. The MLP classifiers were trained in the ALERT-SR.train set. To represent the cluster several alternatives were tested: the feature vector of one of the cluster elements, an average of all elements feature vectors. The first alternative gave the best results. We made $N_{clus} = 300$ frames meaning that the worst case scenario for latency is $N_{clus} + N_{acd} = 600$ frames.

	PER clus	PER spkr	DER
Speaker clustering	15.2	30.2	31.6

Table 4.11: SC results in COST278-BN database.

Results for the evaluation in the COST278-BN corpus are summarized in Table 4.11. Relative to version 1.0 (Table 4.5) we see a significant improvement in DER. Comparing with our COST278 partners we now have slightly worse DER most likely still due to a higher number of cluster per speaker.

4.5 Audio Pre-processor versions 3.0 and 4.0

The new Version of our APP was developed with the intent of reducing the delay for a better online operation, appropriate to work in a real-time online TV subtitling. Another goal was to reduce the complexity, especially in the ACD module while maintaining or increasing the performance. This was specially relevant for European Portuguese BN media (the type of news shows the system has to work on). Apart from these we tried to increase performance even on modules that retained its architecture. A major difference to version 2.0 was the re-introduction of SID (speaker identification) to provide better detection of news anchors and enable the use of speaker dependent acoustic models in the recognition.

4.5.1 Acoustic Change Detection

Version 2.0 ACD module was complex and consequently hard to tune. For the new versions a simpler yet powerful approach was developed (Figure 4.17). The idea was to take advantage of the probabilistic output of the SNS module to determine the start of speech and non-speech segments. To acomplish this the SNS MLP output is smoothed using a median filter with a small window (t_{median} tipically 0.5 seconds). This smoothed signal is thresholded ($Threshold_{Hi} = 0.80$, $Threshold_{Lo} = 0.20$) and analyzed using a time interval t_{Min} of 0.8 seconds by a Finite State Machine (FSM). This FSM uses 4 possible states (Probable Non-speech, Non-speech, Probable Speech and Speech). If the input audio signal has a probability of speech above $Threshold_{Hi}$ the Finite State Machine is put into “Probable Speech” state. If after t_{Min} interval the average speech probability is above a given confidence value the FSM goes to “Speech” state. Otherwise it goes to “Non-speech” state.

The FSM generates segment boundaries for non-speech segments larger than t_{median} (which is related to the resolution of the median window). Additionally if the non-speech segment is larger than t_{Min} the audio signal is discarded. The t_{Min} value is an open parameter of the system and was optimized in the ALERT-SR.devel set so as to maximize the non-speech detected.

Version 3.0 has a maximum delay of $t_{median}/2 + t_{Min} = 0.5/2 + 0.8 = 1.05$ seconds.

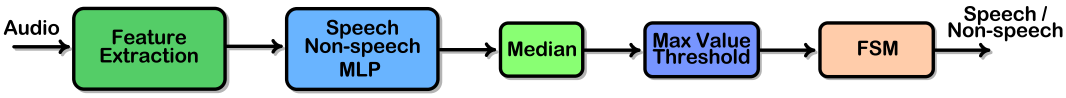

Figure 4.17: ACD / SNS module version 3.0 & 4.0.

Version 4.0 uses t_{Min} =0.85 second window for speech/non-speech decision and a median window of t_{median} =0.25 second. For version 4.0 we introduced a segment extend time parameter for preventing the start/end of a speech segment to close to the actual start/end of the speech which induces recognition errors. This value was set to 0.2 seconds. Also the boundary placed inside a short non-speech segment (between 0.25 and 0.85 seconds) in now placed exactly in the middle of the non-speech segment, again to prevent possible speech recognition errors. Given this, the delay for version 4.0 is $0.25/2 + 0.85 + 0.2 = 1.175$ seconds.

Ac. Change Detector	Recall	Precision	F-measure
version 3.0	74.3	65.3	69.5
version 4.0	80.6	57.6	67.2

Table 4.12: ACD version 3.0 & 4.0 results in COST278-BN database.

Ac. Change Detector	Recall	Precision	F-measure
version 3.0	74.6	65.3	69.7
version 4.0	77.9	63.2	69.8

Table 4.13: ACD version 3.0 & 4.0 results in ALERT-SR.jeval test set.

Table 4.12 and Table 4.13 represent the evaluation results for the COST278-BN database and for the ALERT-SR.jeval test set. Since we are developing a APP system for working in our media monitoring system it is more useful to test its performance using similar news shows to the ones the system will work on. Version 3.0 and 4.0 maintained the good performance of ACD while reduced dramatically its complexity (in terms of system components and therefore simplicity of understanding what is happening and making changes) when compared with version 2.0. Nevertheless this simpler method can have one pitfall which is not detecting speaker turns when there are no short pauses of non-speech in between (in speech overlap for instance).

4.5.2 Speech/Non-speech classification

The SNS classifier uses a MLP with 9 input context frames of 26 coefficients (12th order PLP plus deltas), two hidden layers with 250 sigmoidal units each, and the appropriate number of softmax output units (one for each class) which can be viewed as giving a probabilistic estimate of the input frame belonging to that class.

Version 4.0 has a slightly larger MLP (with two hidden layers of 350 units) and was trained with more data. Specifically examples collected from speech false alarms (jingles and other relevant non-speech segments observed in SSNT-BN data).

Speech/Non-Speech	% Speech	% Non-speech	CER
version 3.0	1.8	29.5	3.7
version 4.0	2.1	14.2	2.9

Table 4.14: SNS version 3.0 & 4.0 results in COST278-BN database.

Speech/Non-Speech	% Speech	% Non-speech	CER
version 3.0	2.1	10.9	2.8
version 4.0	1.7	13.7	2.3

Table 4.15: SNS version 3.0 & 4.0 results in ALERT-SR.jeval test set.

Table 4.14 and Table 4.15 represent the evaluation results for the two test sets we have been using. We see that the SNS module has a very good performance comparable with current state of the art results reported in [Tranter and Reynolds, 2006] as having a CER around 3%. Comparing version 3.0 and 4.0 SNS classification modules with the algorithms developed by our partners in the COST278 [Zibert et al., 2005] (refer back to Figure 4.7) that now we are placed in the first overall position.

4.5.3 Gender Detection

The GD classifier, is similar in architecture to version 2.0 (an MLP architecture with 9 input context frames of 26 coefficients (12th order PLP plus deltas)) except that it has two hidden layers with 250 sigmoidal units each. Classification rates obtained during the training were around 91%. Both versions use the first 300 frames to calculate the speakers gender but version 4.0 discards the first 0.2 second frames to compensate for the segment extend time of ACD / SNS module. These 20 feature frames will probably contain non-speech which is not useful for determining the speakers gender. Given this, version 3.0 has a maximum delay of 3 seconds and version 4.0 of 3.2 seconds.

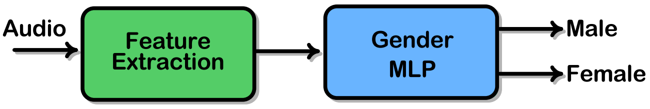

Figure 4.18: GD module version 3.0 & 4.0

Gender	% Male	% Female	CER
version 3.0	2.4	13.2	6.4
version 4.0	1.9	12.0	5.6

Table 4.16: GD version 3.0 & 4.0 results in COST278-BN database.

Gender	% Male	% Female	CER
version 3.0	2.6	2.2	2.5
version 4.0	2.3	1.9	2.2

Table 4.17: GD version 3.0 & 4.0 results in ALERT-SR.jeval test set.

Table 4.16 and Table 4.17 represent the evaluation results for the two test sets we have been using. In the case of ALERT-SR.jeval the performance of GD module is similar to the state of the art published results for which the CER is around 2% [Tranter and Reynolds, 2006]. In the case of the COST278-BN database results can justified with the high classification error rates in Female data (some news shows have Female CER above 50% apparently because of a somewhat lower pitched european female voices).

4.5.4 Background Conditions classification

Background Conditions classification has the same architecture that the GD module with a 3 seconds maximum delay. Background is being used for Topic Segmentation purposes (a later module in our pipelined media monitoring system).

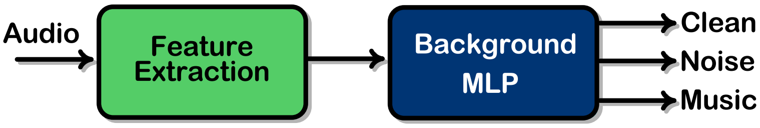

Figure 4.19: BC module version 3.0 & 4.0.

Background	% Clean	% Music	% Noise	CER
version 3.0	41.5	70.3	22.8	36.4
version 4.0	—	—	—	—

Table 4.18: BC version 3.0 results in COST278-BN database.

Background	% Clean	% Music	% Noise	CER
version 3.0	22.1	34.5	11.1	15.3
version 4.0	—	—	—	—

Table 4.19: BC version 3.0 results in ALERT-SR.jeval test set.

Table 4.18 and Table 4.19 represent the evaluation results for the two test sets we have been using. BC classification besides being a rather difficult task is not commonly found in current state of the art audio diarization systems. Nevertheless our CER is still high and we are working on better models.

4.5.5 Speaker Clustering

Version 3.0 and 4.0 Speaker Clustering module works in the following way (represented in Figure 4.20): after the ACD / SNS detector signals the existence of a new speech boundary and the GD classification module determine the gender, the first 300 frames (at most) of the segment are compared with all the clusters found so far, for the same gender. The segment is merged with the cluster with the lowest distance, provided it falls bellow a predefined threshold (stop criterion). 12th order PLP plus energy but without deltas was used as feature extraction. The distance is computed using the “standard” Bayesian Information Criterion (BIC) with full covariance matrices and with two thresholds λ and α. As in SC module of Version 1.0 the second threshold, α, is a cluster adjacency term which favours clustering together consecutive speech segments. Empirically if the speech segment and the cluster being compared are adjacent (closer in time) the probability of belonging to the same speaker must be higher. The thresholds were tuned in the ALERT-SR.devel test set in order to minimize DER ($\lambda = 2.75, \alpha = 1.40$). Version 4.0 has a maximum delay of 320 frames to compensate the non-speech extend segment time of ACD/SNS module. Also version 4.0 works in parallel with GD and SID to keep the delay at the 320 frames. Since SC depends on GD and SID (male and female) decisions, version 4.0 has two SC modules in parallel (one for each gender). Final decision of cluster ID is only taken after GD, SID male, SID female and SC male and SC female have finished computing. Creating / updating clusters is also done after. Although this architecture is more CPU intensive that version 3.0, in practice keeping the delay is more important.

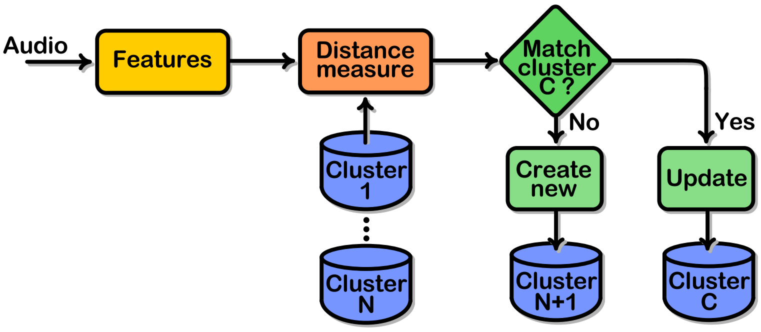

Figure 4.20: SC module version 3.0 & 4.0.

Clustering	PER clus	PER spkr	DER
version 3.0	11.1	20.9	27.6
version 4.0	12.6	16.4	28.4

Table 4.20: SC version 3.0 & 4.0 results in COST278-BN database.

Our clustering module has lower performance than the best algorithms for which DER results around 9% have been reported [Zhu and et al., 2005], obtained with state of the art speaker identification techniques like feature warping and model adaptations. Although

Clustering	PER clus	PER spkr	DER
version 3.0	8.7	22.0	26.1
version 4.0	8.3	24.4	25.2

Table 4.21: SC version 3.0 & 4.0 results in ALERT-SR.jeval test set.

very efficient these techniques are hard to setup (thousands of hours of speech data are needed for training reliable models) and are very slow running ([Gales et al., 2007] reports using a 36% DER APP module for their 1xRT speech recognition system because their best APP module which has 9% DER does not meet the 1xRT time constrain). Compared with this situation our version 4.0 APP module works in 0.1xRT and has a much lower DER!

4.5.6 Speaker Identification

Anchors introduce the news and provide a synthetic summary for the story. Normally this is done in studio conditions (clean background) and with the anchor reading the news. Anchor speech segments convey all the story cues and are invaluable for automatic topic indexation and summary generation algorithms. Besides anchors there are normally some important reporters who usually do the main and large news reports. This means that a very large portion of the news show is spoken by very few (recurrent) speakers. It is therefore more important to get the main speakers complete and correct than to accurately find speakers who do not speak much. Since we are daily monitoring a predefined news show from RTP station we wanted to investigate if using appropriate speaker models for the main station speakers would improve clustering performance.

For this purpose a Seaker ID (SID) module was built. Figure 4.21 shows the SID module architecture. It is composed by feature extraction (26 PLP coefficients) and a MLP classifier. Speaker models were built for the male and female news anchor speakers. In version 4.0 we have two MLP classifiers, one for the two male news anchors and the other for the two more important female news anchors. Version 3.0 covered only one male anchor and

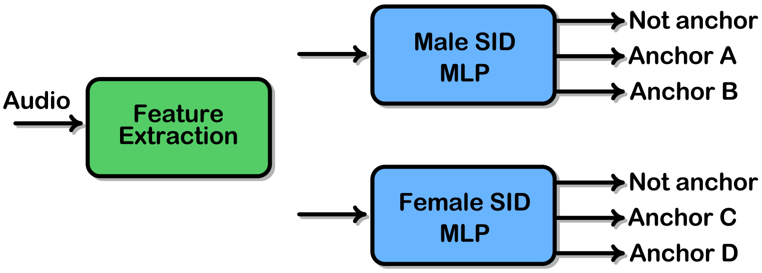

Figure 4.21: SID module of APP version 3.0 & 4.0.

one female anchor. Version 4.0 was trained with more data and data from SSNT-BN because the newer anchors are not present in ALERT-SR corpus. Maximum delay is equal to GD, 300 frames used to calculate the probability of the segment belonging to one of the news anchors.

SID + Clustering	PER clus	PER spkr	DER
version 3.0	9.0	20.8	24.8
version 4.0	11.9	14.7	20.6

Table 4.22: SID + Clustering version 3.0 & 4.0 results in ALERT-SR.jeval test set.

Table 4.22 summarizes the diarization results in the ALERT-SR.jeval test set for version 3.0 and 4.0. We can see that the use of SID prior to clustering brought a significant DER reduction.

4.6 Summary

This chapter described the development of our Audio Pre-Processing module. The first version was developed for aiding the annotation of ALERT-SR corpus. The second version was already stream-based and had better performance in COST278-BN database while having a low latency. It relied heavily on model-based techniques. The third and fourth versions improved the overall performance specially in SNS and GD and SC modules which have a performance comparative with state of the art results. It also introduced SID models that were responsible for a significant DER reduction.

In terms of diarization, better results (below 20%) are reported for agglomerative clustering approaches [Tranter and Reynolds, 2006]. This type of off-line processing can effectively perform a global optimization in the search space and will be less prone to errors when joining together short speech segments than the on-line clustering approach we have adopted. This approach not only is doing a local optimization of the search space but also the low latency constraint involves comparing a very short speech segment with the clusters found so far.

The best speaker clustering systems evaluated in BN tasks achieve DER results around 9% by making use of state of the art speaker identification techniques like feature warping and model adaptation [Zhu and et al., 2005]. Such results, however, are reported for BN shows which typically have less than 30 speakers, whereas the BN shows included in the ALERT-SR.jeval test set have around 80. Additionally no information is given regarding the time taken to produce the state of the art results published. [Gales et al., 2007] cites that although their best diarization system has around 9% DER, when building a 1xRT ASR system they had used an APP module which had 38.6% DER.

We evaluated the performance of the system components using the COST278-BN database and the ALERT-SR.jeval test set that is being used as test bed for comparison of different algorithms. The COST278-BN database has the great advantage of simplifying direct comparisons using the same evaluation tools and protocols without the need for implementing all state of the art algorithms.

Also it is useful to compare our results against results from two evaluation campaigns for the English (NIST RT04F) and French (ESTER phase II). Looking at the European Portuguese BN test data (described in Chapter 2) we can understand the differences in the results obtained. This test data has significantly more speakers (longer duration) and diverse acoustic conditions than NIST RT04F and ESTER test sets. According to [Zhu and et al., 2005] the error rate more than doubles when comparing one news show with 20 min with another that lasts 1 h. Again this source clearly indicates that higher DER are correlated with longer shows that have more different speakers. Our ALERT-SR.jeval test set has 14 news shows with about 1 hour long and has many different speakers and background conditions compared with NIST RT-04F (half an hour long) and ESTER phase II evaluation shows. Statistics for the number of speakers in our jeval test set: min = 41, max = 81, average = 57. Compared with RT-04F which has always less than 27. This helps to explain why WER is higher.

Chapter 5

Prototype Implementation

One of the goals of our laboratory was to build a prototype demonstration media monitoring system for the European Portuguese language. This chapter describes in detail the implementation of our media monitoring prototype. The present implementation is focused on demonstrating the usage and features of this kind of system. Our prototype is running daily since May 2002 with success processing the 8 o'clock evening news of our public TV broadcast company (RTP).

Our media monitoring prototype system has a set of news shows to monitor from different BN TV stations and has a set of registered users each one with a profile regarding the news topics that are of his/her interest. The media monitoring system processes a news show and compares the topics generated by the stories in the news show against the used profiles. It then sends alert emails to the users with the title, short summary and video link to the relevant news detected.

Our media monitoring prototype system [Meinedo and Neto, 2003b] is composed by three main blocks: a CAPTURE block, responsible for capturing and converting each of the news shows defined to be monitored, a PROCESSING block, responsible for generating all the relevant mark-up information for each news show, and a SERVICE block, responsible for the user and database management interfaces. Figure 5.1 represents our media monitoring block diagram.

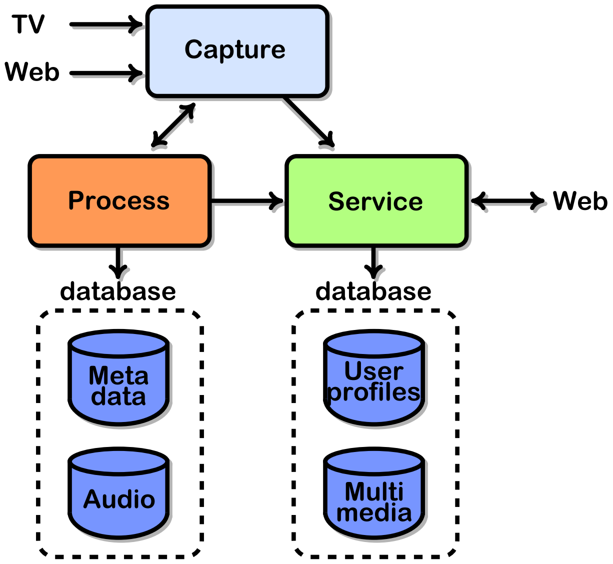

Figure 5.1: Media monitoring prototype block diagram

5.1 CAPTURE block

The CAPTURE block is responsible for recording and converting each of the news shows defined to be monitored. Figure 5.2 represents the CAPTURE block diagram. The first module ("Schedule recordings") has a list of news shows which are to be processed. The list indicates the name of the news show and the name of the TV station where the show is transmitted. The "Schedule recordings" module then downloads from the TV station web site an html with the daily time schedule. It parses the html file to retrieve the expected starting and ending time of the news show. It is frequent that the actual news show duration is larger than what had been advertised in the TV station time schedule (possibly due to live broadcasts or interviews). To ensure that the complete news show is captured the module schedules the recording to start a little earlier (1 minute before) than announced and records much more time after the advertised ending time (20 minutes after). To trim the unwanted recorded portions that do not belong to the news show our media monitoring prototype uses a jingle detection module. This module which is part of the PROCESSING block (see section 5.2.1 ahead) is responsible for detecting the actual start and end times of the news show.

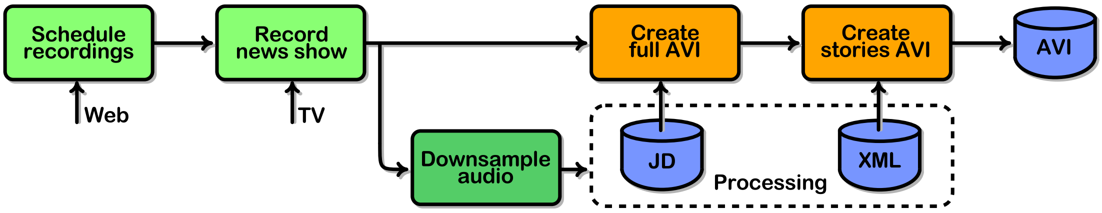

Figure 5.2: Prototype CAPTURE block diagram.

The CAPTURE "Record news show" module records the specified news show at the defined time using a TV capture board (Pinnacle PCTV Pro) that is connected to the cable TV network. The recording produces two independent streams: a MPEG-2 video stream and a uncompressed, 44.1 kHz, mono, 16 bit audio stream. Since our prototype currently works in offline mode, these two streams are stored on disk in separate files. After the recording finishes the generated audio stream file is downsampled to 16 kHz which is the sampling rate used in all our speech processing modules. The original 44.1 kHz audio stream file will be used together with the video stream file to generate a new video of the complete news show. When the audio downsampling is completed a "ready" flag signal triggers the PROCESSING block. After the PROCESSING block sends back jingle detection information (see section 5.2.1 ahead) and changes the signal flag from "ready" to "proc" the CAPTURE block starts to multiplex the recorded video and audio streams together to produce the video of the complete news show. This is represented in Figure 5.2 by module "Create Full AVI". This module uses the jingle detection information to cut out the recorded portions that do not belong to the news show (commercials). The trimmed and multiplexed video and audio file is stored in AVI format and has MPEG-4 video and MP3 audio. The CAPTURE block waits again for information coming from the PROCESSING block.

When the PROCESSING block finishes all processing it sends back to the CAPTURE block the resulting XML file (see section 5.2 ahead) and changes the signal flag from "proc" to "proc2". Using the XML information the "Create stories AVI" module from

CAPTURE block starts to generate AVI video files for each news story. These individual story AVIs are cutted out from the full news show AVI and have less video quality (more compression) which is appropriate for streaming to portable devices like PDAs or mobile phones. All the AVI video files generated by the CAPTURE block are then sent to the SERVICE block for additional conversion and storage (see section 5.3 ahead).

5.2 PROCESSING block

The PROCESSING block is responsible for automatically generating all the relevant mark-up information for the news show being processed. It takes as input the 16 kHz audio stream recorded by the CAPTURE block and generates as output an XML file containing the metadata information from 6 modules represented in the PROCESSING block diagram of Figure 5.3.

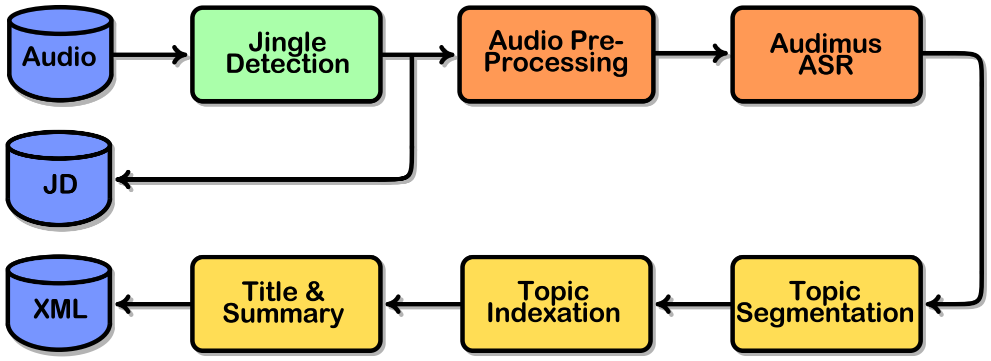

Figure 5.3: Prototype PROCESSING block diagram.

First the recorded audio stream file is processed by the “Jingle Detection” (JD) module which is responsible for identifying the exact start and end times of the news show. This is accomplish by the identification of special music patterns used in BN shows to indicate the beginning and end of the show. These patterns are know as “jingles” that is a term used in marketing for a music pattern which is used for drawing the listener attention. The JD module also identifies other audio portions that are not relevant like news fillers and detects commercial breaks inside the news show. Again this is possible because these events are signalled by jingles. Using the detected time instants a new audio stream file is generated containing only the relevant contents of the news show. Also the JD information is sent back to the CAPTURE block for generating the full news show AVI. The new audio stream file is fed through the Audio Pre-Processing (APP) module whose detailed description in given is Chapter 4. This module is responsible for the segmentation of the audio into acoustically homogeneous regions, for classification of those segments according to speech and non-speech intervals, background conditions, speaker gender and for identifying all segments uttered by the same speaker. Each audio segment that was marked by the Audio Pre-processor as containing speech (transcript segment) is then processed by the Automatic Speech Recognition (AUDIMUS ASR) module which whose detailed description was given in Chapter 3. The Topic Segmentation (TS) module groups several transcript segments into news stories. For each story the Topic Indexing (TI) module generates a classification about the contents of the story according to a hierarchically organized thematic thesaurus [Amaral and Trancoso, 2003b, Amaral and Trancoso, 2003c, Amaral and Trancoso, 2003a]. Finally there is a module for generating a synthetic Title and Summary for each news story. Together with the Topic Indexing information the user has a better idea about the news story content [Neto et al., 2003a, Trancoso et al., 2004]. At the end of the PROCESSING block an XML file containing all the relevant information that these 6 modules were able to extract is generated according to a DTD specification [Neto et al., 2003b, Neto et al., 2003a]. This XML is sent back to the CAPTURE block for generating individual story AVI files and sent forward to the SERVICE block for storage in the database.

The next subsection describes in detail the Jingle Detection module. The rest of the PROCESSING section evaluates the impact of prior processing modules in later ones. The results presented cover the influence of APP in ASR results and their influence on TD results.

5.2.1 Jingle Detection module

Accurate detection of the news show start and end jingles is crucial to the performance of our media monitoring system. An incorrect identification of the start or end of the news show will probably result in mistakes in latter modules (APP, ASR and TD). The BN shows that are being processed by our media monitoring system have basically four different types of jingles: start jingle which marks the beginning of the news show, end jingle marking the end of the show, publicity jingle either marking the beginning/ending of a commercial break or appearing between commercials and finally the filler jingles. The filler jingles appear when the news anchor is giving emphasis to some news stories that will be developed later in the show or is summarizing the news stories that were covered during the show. In either situation these filler sequences do not convey relevant information and can induce errors in the TD modules because inside these fillers the topic is changing rapidly.

As a prototype demonstrator the JD module of our media monitoring system [Meinedo and Neto, 2004] was tuned for processing jingles from 3 different news shows. The ”Telejornal” which is the main 8 o’clock news show from RTP, the ”Jornal2” which is the 9 o’clock news show from RTP2 station and the ”Jornal Nacional” which is again the main news show (8 o’clock) but from the private station TVI. ”Telejornal” news show is currently characterized by two jingles, one for delimiting the main body of the broadcast news (start, end and fillers) and another to mark the commercials. ”Jornal2” news show has a jingle for the start of the show and a different one for the end. It has no commercial breaks or filler blocks. The private channel news show (”Jornal Nacional”) has four different jingles: start, end, publicity and filler. Figure 5.4 represents a possible time sequence for a news show similar to the ones that our media monitoring system is processing.

Fig 5.4 represents the TVI station News show time sequence and its corresponding jingle events (start and end jingles, the filler jingles, useful content (News) and the other non use-

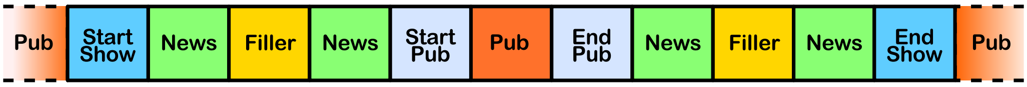

Figure 5.4: News show time sequence and events.

ful portions that will be discarded (publicity)). Normally there are several filler sections. To represent the events transitions of a particular news show we developed appropriate Finite State Models (FSM). Figure 5.5 represents the Finite State Model diagram for one of the news shows being processed (”Jornal2”) which is the one with the simplest structure.

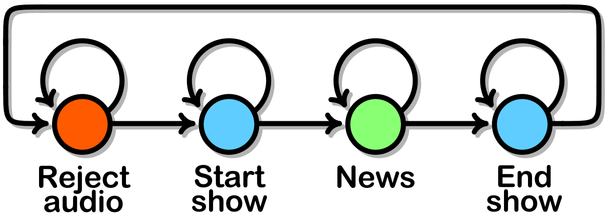

Figure 5.5: “Jornal2” Finite State Model diagram.

In “Jornal2” FSM there is one state for rejecting the audio before the news show starts, one state while the audio belongs to the start jingle, one for recording the useful audio and one for the end jingle. After the end jingle there is a loop back to the “reject audio” state for discarding the audio after the news show ends. In this FSM it is possible to stay in the same state as long as necessary (for instance when recording the news). The FSM also takes into consideration the minimum time duration of the jingles. That is, if a jingle occurred but it lasted less than its pre-defined minimum duration the FSM does not change state. The minimum time duration of an event is useful for preventing false alarms.

The other two news shows being processed by our media monitoring system have a somewhat more intricate FSM which is represented by Figure 5.6 for “Telejornal”. The major difference is the presence of 2 sub-FSM describing the Filler sections and commercial breaks (Pub ‘= Publicity). The Fillers have a state for representing the start of the filler, one for rejecting the filler audio and finally one state for the end of filler jingle. The Pub sub-FSM has 3 states: start / end of the commercial break and one for rejecting the

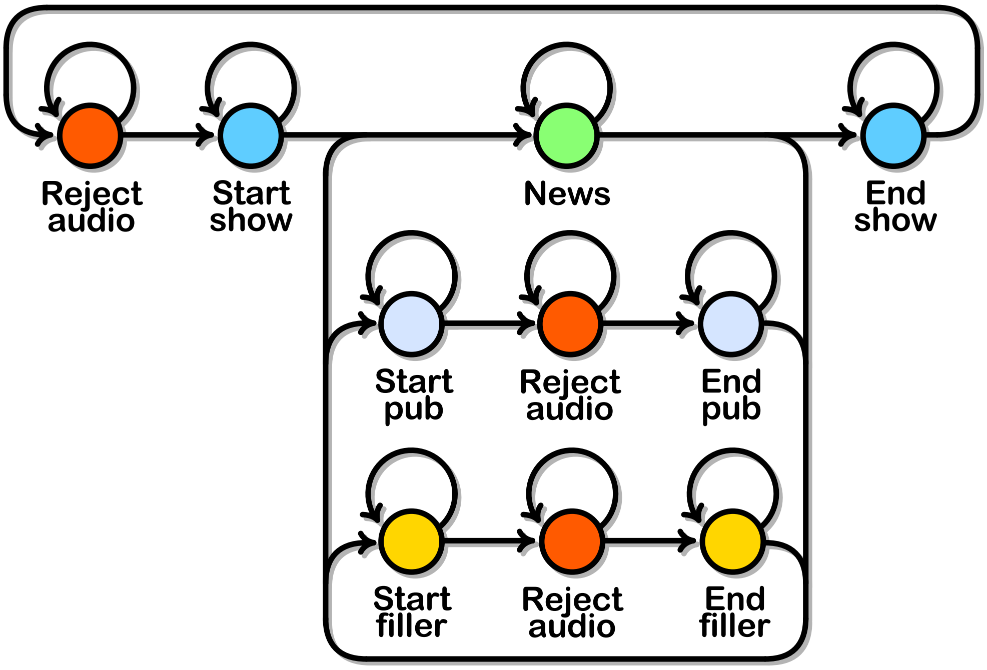

Figure 5.6: “Telejornal” Finite State Model diagram.

commercials. Additionally there can be a state for consuming the Publicity jingle that appears between commercials. The “Jornal Nacional” news show uses the same FSM as “Telejornal” but is somewhat easier to process because it has a different jingle for each event while “Telejornal” uses the same jingle music for start, end and filler events.

The block diagram of our JD module is represented in Figure 5.7 and includes 5 main components. The first one extracts PLP (Perceptual Linear Prediction) [Hermansky et al., 1992] features from the incoming audio signal. It uses a sliding window of 20 ms, which is updated every 10 ms and extracts 26 parameters per frame (12th order plus log energy plus first order derivatives).

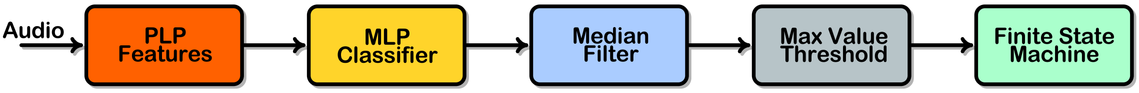

Figure 5.7: Jingle Detector block diagram.

The second component is a pattern classifier that classifies these acoustic feature vectors and is trained to estimate at the output the probability of the given time frame being a certain jingle. Our approach uses Artificial Neural Networks to detect the jingles that mark the start and end of the news show. Neural networks are widely known pattern classifiers that possess good generalization capabilities when correctly trained. But unlike many other pattern classification applications we are not interested in the generalization capabilities of the neural network, as the goal is to detect a pre-determined acoustic pattern (the jingle) and not generalize to similar ones. Our Artificial Neural Network classifier is of the MLP type (Multi-Layer Perceptron). The MLP architecture includes 9 input context frames, two hidden layers with 75 units each and several output units (one for each type of jingle). The total number of parameters (weights) is less than 24000 which is rather small. The output of this classifier is then smoothed by a median filter with a 31 frame window and compared to a predetermined threshold value. After some adjustments in the training set this threshold was set to 0.9, that is, it only considers that the jingle occurred if the average frame probability is higher than 90% in the time interval specified by the minimum jingle duration. This minimum duration was set to 0.8 seconds for all jingles causing the total maximum delay to be half the median window plus the minimum jingle duration, that is, slightly below 1 second. The last component is a Finite State Machine that uses the models represented in Figures 5.5 and 5.6. This component uses the jingle events detected for taking a decision regarding the FSM diagram transitions and for starting/stoping the recording of the useful audio signal.

A different MLP classifier was trained for each news show being processed. These 3 MLPs are responsible for classifying all the jingles that are used for that particular news show. The “Jornal2” MLP detects 2 different jingles (start and end), the “Telejornal” MLP classifier detects 4 different jingles (start, end, filler and pub) the “Jornal Nacional” classifier also detects 4 different jingles. These 3 jingle MLP classifiers were trained using the back-propagation and gradient descent algorithm [Almeida, 1996] in stochastic mode with non-adaptive training step. The stop criterion was the mean square error in the cross validation set. We also used output error weighting factors in order to balance class a priori distributions [Lawrence et al., 1998].

As we can see from Table 5.1, an adequate total number of training frames was chosen in

	Number of jingles		Total number of
Jingle	Training	Validation	Frames
“Jornal2” start	20	5	16047
“Jornal2” end	5	2	3694
“Telejornal” start/end	18	5	12868
“Telejornal” filler	45	15	18152
RTP publi	28	6	11552
“Jornal Nacional” start	7	2	6599
“Jornal Nacional” end	8	2	12313
“Jornal Nacional” filler	26	8	8972
TVI publi	10	3	3617
Reject patterns	26	3	79726

Table 5.1: Amount of training material for each jingle.

order to have a high training pattern to MLP weight ratio (at least 10 times more training patterns than MLP weights). The cross validation set was chosen in order to have about 20% of the total number of training patterns. The frame CER (Classification Error Rate) obtained after training was well below 5% in the cross validation set.

The evaluation of the 3 distinct jingle detectors was done with 3 test sets, one for each different type of news show that our media monitoring system is processing. Each of these test sets is composed by one news show which typically lasts almost an hour (half an hour in the “Jornal2” case). These test news shows are not from the ALERT-SR corpus neither from the COST278-BN corpus since unfortunately TV station jingles change at least once or twice a year. These test news shows were chosen from recent recorded emission and manually labelled for this evaluation. We considered as evaluation criteria the frame Jingle Error Rate (% JER) and the frame News Error Rate (% NER) which represents respectively the percentage of jingle frames incorrectly classified and the percentage of news (useful content) frames incorrectly classified. Table 5.2 shows our results including the time taken to process the news shows, expressed in terms of times real time (x RT).

These results were obtained in a 2.66 GHz Pentium 4 PC. Although the relatively high JER, that is, the algorithm is not capable of correctly tagging all the frames that belong to the jingles, the NER (useful content error rate) is very small. In the ”Telejornal” and in the

News show	% JER	% NER	x RT
“Telejornal”	20.7	5.2	0.02
“Jornal2”	10.8	2.0	0.02
“Jornal Nacional”	15.4	4.1	0.03

Table 5.2: Jingle Detection evaluation results.

“Jornal Nacional” news shows the algorithm failed to classify some of the filler segments and marked them as containing useful audio. This explains the higher NER. Sometimes these filler jingles are shorter than the minimum duration or played with volume fade-in which makes their identification very difficult.

5.2.2 ASR results with manual and automatic prior processing

The APP module automatically determines which audio segments are sent to the ASR module for transcription. Since the APP module is not perfect we wanted to evaluate its impact on speech recognition performance. Table 5.3 presents the word error rate (WER) results on the ALERT-SR.jeval test set for the F0 focus conditions and for all conditions. We conducted two different experiments. First the test set was recognized using the hand labelled reference classifications and sentence segmentation boundaries (this is indicated in Table 5.3 as “manual” APP). Then the test set was recognized using our APP classifications and segmentation information (indicated as “automatic” APP).

	% WER	
APP	F0	All
Manual	11.3	23.5
Automatic	11.5	24.0

Table 5.3: Automatic Speech recognition results.

The performance is comparable in both experiments with only 0.5% absolute increase in WER. This increase can be explained by Speech / Non-speech classification errors, that is, word deletions caused by noisy speech segments tagged by APP as non-speech, and word insertions caused by noisy “silence” segments marked by APP as containing speech. The other source for errors, and the most relevant one, is related to the differences between manual and automatic sentence-like unit (SU) boundaries. Since the APP tends to create larger than "real" SUs, the problem seems to be in the language model which is introducing erroneous words (mostly function words), connecting different SUs.

5.2.3 TS results with manual and automatic prior processing

This work is part of my colleague's Rui Amaral PhD report. The goal of TS module is to split the BN show into the constituent stories. This may be done taking into account the characteristic structure of BN shows [Barzilay et al., 2000]. They typically consist of a sequence of segments that can either be stories or fillers. The fact that all stories start with a segment spoken by the anchor, and are typically further developed by out-of-studio reports and/or interviews is the most important heuristic that can be exploited in this context. Hence, the simplest TS algorithm is the one that starts by defining potential story boundaries in every transition non-anchor / anchor. In the next step, the algorithm tries to eliminate stories that are too short, because of the difficulty of assigning a topic with so little transcribed material. In these cases, the short story segment is merged with the following one with the same speaker and background. Other heuristics are also adopted to avoid too long stories spoken only by the anchor, which may in fact include more than one story without further developments. The identification of the anchor is done on the basis of the speaker clustering information, as the cluster with the largest number of turns. A minor refinement was recently introduced to account for the cases where there are two anchors (although not present in the ALERT-SR.jeval test set). The evaluation of the Topic Segmentation was done using the standard measures Recall (% of detected boundaries), Precision (% of marks which are genuine boundaries) and F-measure (defined as $2RP/(R+P)$). Table 5.4 shows the TS results, using the Recall, Precision, and F-measure metrics, as well the metric adopted in the 2001 Topic Detection and Tracking benchmark NIST evaluation, with the same cost values of miss and false alarms [Group, 2001]. These results, together with the field trials we have conducted [Trancoso et al., 2003], show that boundary deletion is a critical problem. In fact, our very simple TS algorithm has several pitfalls: it fails when all the story is spoken by the anchor, without further reports or interviews, leading to a merge with the next story; ii) it fails when the filler is not detected by a speaker / background condition change, also leading to a merge with the next story (19% of the program events are fillers); iii) it fails when there is a special anchor for a part of the broadcast (i.e. sports anchor), although in this case one could argue that all the stories are about the same generic topic; iv) it fails when the anchor(s) is not correctly identified.

APP	ASR	Recall	Precision	F-measure	Cost
Manual	Manual	79.0	60.3	67.2	0.78
Manual	Auto	76.6	60.3	66.3	0.75
Auto	Auto	70.2	56.4	61.6	0.66

Table 5.4: Topic Segmentation results.

5.2.4 TI results with manual and automatic prior processing

This work is part of my colleague's Rui Amaral PhD report. Topic identification is a two-stage process, that starts with the detection of the most probable top-level story topics and then finds for those topics all the second and third level descriptors that are relevant for the indexation. For each of the 22 top-level domains, topic and non-topic unigram language models were created using the stories of the ALERT-TD corpus which were pre-processed in order to remove function words and lemmatize the remaining ones. Topic detection is based on the log likelihood ratio between the topic likelihood $p(W/T_i)$ and the non-topic likelihood $p(W/\overline{T_i})$. The detection of any topic in a story occurs every time the correspondent score is higher than a predefined threshold. The threshold is different for each topic in order to account for the differences in the modelling quality of the topics. In the second step, we count the number of occurrences of the words corresponding to the domain tree leafs and normalize these values with the number of words in the story text. Once the tree leaf occurrences are counted, we go up the tree accumulating in each node all the normalized occurrences from the nodes below [Gelbukh et al., 2001]. The decision of whether a node concept is relevant for the story is made only at the second and third upper node levels, by comparing the accumulated occurrences with a pre-defined threshold. In order to conduct the topic indexation experiments we started by choosing the best threshold for the word confidence measure as well as for the topic confidence measure. The tuning of these thresholds was done with the development corpus in the following manner: the word confidence threshold was ranged from 0 do 1, and topic models were created using the correspondent topic material available. Obviously higher threshold values decrease the amount of automatic transcriptions available to train each topic. Topic indexation was then performed in the development corpus in order to find the topic thresholds corresponding to the best topic accuracy (91.9%). The use of these confidence measures led to rejecting 42.0% of the original topic training material. Once the word and topic confidence thresholds were defined, the evaluation of the indexation performance was done for all the stories of the ALERT-SR.jeval test set, ignoring filler segments. The correctness and accuracy scores obtained using only the top-level topic are shown in Table 5.5, assuming manually segmented stories. Topic accuracy is defined as the ratio between the number of correct detections and the total number of topics, and topic correctness as the ratio between the number of correct detections minus false detections (false alarms) and the total number of topics. The results for lower levels are very dependent on the amount of training material in each of these lower level topics (the second level includes over 1600 topic descriptors, and hence very few material for some topics). When using topic models created with the non-rejected keywords, we observed a slight decrease in the number of misses and an increase in the number of false alarms. We also observed a slight decrease with manual transcriptions, which we attributed to the fact that the topic models were built using ASR transcriptions.

These results represent a significant improvement over previous versions [Amaral and Trancoso, 2004], mainly attributed to allowing multiple topics per

APP	ASR	Correctness	Accuracy
Manual	Manual	91.5	91.3
Manual	Auto w/o conf	93.8	91.5
Manual	Auto w/ conf	94.1	91.7
Auto	Auto w/ conf	93.9	91.4

Table 5.5: Topic indexation results.

story, just as in the manual classification. A close inspection of the table shows similar results for the topic indexation with auto or manual APP. The adoption of the word confidence measure made a small improvement in the indexation results, mainly due to the reduced amount of data to train the topic models. The results are shown in terms of topic classification and not story classification. Whereas one could find comparable results for topic segmentation in the TDT2001 evaluation program [Group, 2001], the topic indexation task has no parallel, because it is thesaurus-oriented.

5.3 SERVICE block

The SERVICE block represented in Figure 5.8 is responsible for the user and database management interfaces. It receives all the AVI video files created in the CAPTURE block and converts them into Real Media (RM) format which is the appropriate format for streaming video and audio over the web. The SERVICE block is also responsible for parsing and loading the XML file generated in the PROCESSING block into the media monitoring system database. From the XML information it creates SMIL and Real Text (RT) files for each of the AVI video files. These SMIL and RT will be used when streaming the video together with the transcribed subtitles.

The SERVICE block is also responsible for running the web video streaming server (Helix Server software) and running the web pages server for the user interface (`http://ssnt.l2f.inesc-id.pt`). It manages the user profiles in the database and sends alert email messages to the registered users resulting from the match between

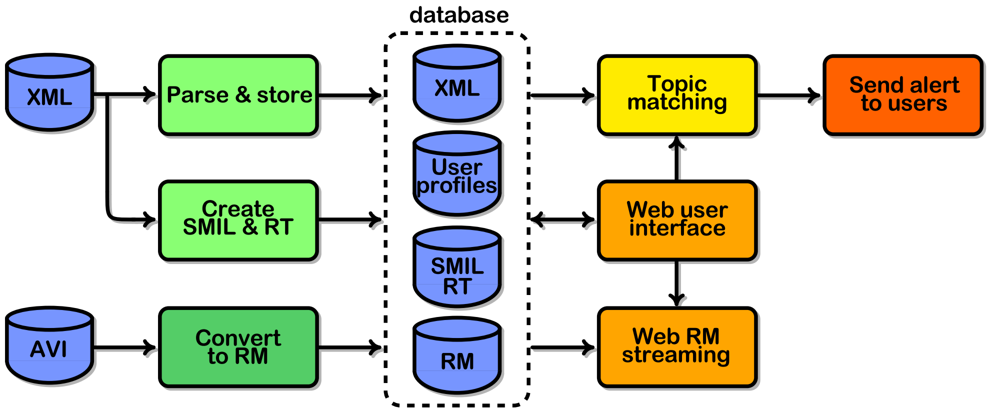

Figure 5.8: Prototype Service block diagram.

the news show information and the users profiles [Neto et al., 2003b, Neto et al., 2003a].

5.4 Summary

This chapter described the prototype implementation of our media monitoring system. Built from in-house developed component modules (Jingle Detection, Audio Pre-Processing, Automatic Speech Recognition, Topic Segmentation and Indexing) our prototype is fully functional and is processing every day the 8 o'clock evening news show of the Portuguese public broadcast company RTP and sending alert messages for registered users (`http://ssnt.l2f.inesc-id.pt`).

We described in detail each block that composes the media monitoring prototype: CAPTURE, PROCESSING and SERVICE blocks. In the PROCESSING block we described the Jingle Detection module which plays a very important part because it is responsible for the correct identification of relevant news show audio by detecting start and end jingles. Furthermore it detects non useful commercial breaks and filler segments. The JD evaluation results are solid although not perfect especially with the correct detection of some filler segments. The algorithm is very fast taking only a fraction of real time. Also in the PROCESSING block we investigated the impact of automatic prior modules in latter ones. That is, the impact of APP errors in ASR performance and the impact of APP and

ASR errors in TD performance. These tests were conducted using the ALERT-SR.jeval test set. The impact of APP errors on the ASR performance is small (0.5% absolute) when compared with the manual references. The greatest impact of APP errors is in terms of Topic Segmentation given the heuristically-based approach that is crucially dependent on anchor detection precision (APP speaker clustering). The ASR WER results are worse than the ones that are quoted for other languages, such as English (less than 16% with real-time performance [Matsoukas et al., 2005]), a fact that can be partly attributed to the reduced amount of BN training data for European Portuguese (51 h compared with over 2000h reported for English). The ASR errors seem to have very little impact on the performance of the two next modules (TS and TI), which may be partly justified by the type of errors (e.g. errors in function words and in inflected forms are not relevant for Topic Indexation purposes).

Chapter 6

Conclusions and Future Work

This chapter presents some conclusions for the most important parts of the thesis work. It also proposes and discusses some possible actions to be addressed by our future work. Broadcast News speech processing is a very difficult task due to a number of well known factors. Some of the BN speech problems can and should be addressed by our future work in order to make our media monitoring system more robust and have better performance. The concluding remarks and possible future directions were divided into sections, BN corpora, Audio Pre-Processing, Automatic Speech Recognition and Prototype Implementation.

6.1 BN corpora

The collection of appropriate BN corpora was fundamental for the developement of our BN media monitoring system and all its constituent modules. This was a big effort for our laboratory specially in terms of man power and time. These resources, the ALERT-SR, ALERT-TD, COST278-BN, WEBNEWS-PT and SSNT-BN (both growing daily) corpora compose a solid resource platform for building other systems and expanding our current systems performance.

6.2 Audio Pre-Processing

Version 1.0 was our first APP module and was built to help the automatic annotation process of the BN ALERT corpora and to serve as the first approach to APP in our media monitoring system. This version was rudimentary but got the job well done. Nevertheless the participation in COST278 evaluation campaign showed us that we needed an APP module with better performance. Version 2.0 was developed to improve its performance and to change the operation mode from offline to online (stream-based). Version 3.0, our latest APP module introduced better SID models and has very good performance. GD and SID are used for switching between ASR gender and speaker dependent acoustic models.

Although our main BN corpus, the ALERT-SR, enabled us to build powerful APP systems we detected hand labelled tags which are incorrect. For instance, we detected many clean speech segments marked as containing background noise. This training data contamination certainly lowers the performance of our BC modules. Another example is incorrect acoustic change point boundaries. We propose two solutions for this problem: to increase the quality of our current training data we need to do an automatic reclassification using confidence scores from our current models and retrain new models using the new classifications. After some iterations we will have a better more precise reference training set. The other proposed solution is to increase the training data by using the SSNT-BN database discussed in chapter 2 of daily recorded news shows. Again, automatic and unsupervised classification of this new data will be necessary.

6.3 Automatic Speech Recognition

AUDIMUS.media was the first European Portuguese BN ASR system. We observed a enormous evolution with the availability of more training data from ALERT-SR corpus, bigger MLPs in the acoustic models, better LMs and a much better decoder using WFSTs.

This BN ASR system has a higher WER than the state of the art ASR systems for English and French but is able to drive sucessfuly our media monitoring system. Nevertheless we continued to investigate ways to improve the recognition performance. One of them was to increase the amount of training data. ALERT-SR training set has less than 50 hours of speech. Several sources report a performance degradation between 14 to 18% when using less than 100 hours of speech training data [Wessel and Ney, 2001]. State of the art ASR systems for English and French were trained at least with 190 h of manually labeled training data. Given this we clearly would benefit from more BN training data. The solution for increasing our training set was to use our SSNT-BN database of automatically collected BN news shows. We developed an unsupervised selection and annotation process using our current APP and ASR systems in order to have more training data. Another strategy for improving our ASR performance was to build gender dependent acoustic models. This was possible with the automatically collected and transcribed SSNT-BN database. A more effective approach to reduce WER was to build speaker adapted acoustic models for certain important speakers like news anchors which have a lot of speech. We built anchor adapted models for our main news anchors and reduced significantly the WER in the ALERT-SR.jeval test set.

A possible future direction for improving ASR performance is to develop a better feature extraction module combining the outputs of different features using MLPs (similar techniques employed by SRI and ICSI speech recognition systems [Zhu et al., 2005b, Zhu et al., 2005a]). The authors report promissing results for this technique.

6.4 Prototype Implementation

The prototype implementation of our media monitoring system was built from inhouse developed component modules (Jingle Detection, Audio Pre-Processing, Automatic Speech Recognition, Topic Segmentation and Indexing). It is fully functional and is processing every day the 8 o'clock evening news show of the Portuguese public broadcast company RTP and sending alert messages for registered users (http://ssnt.l2f.inesc-id.pt).

Appendix A

LDC Hub4 Transcription Conventions

For transcribing our speech corpora we adopted the LDC Hub4 Broadcast News speech transcription conventions [LDC-Hub4, 2000]. Additionaly we established that the following transcription rules needed clarification:

- Channel and fidelity attributes of speaker turns
- Speech utterance segmentation
- Silences inside speaker turns
- Labeling of section blocks
- Identification of jingle segments
- Marking of foreign language utterances
- Transcription of interjections and non-lexemes

The major speaker turn attributes were channel (studio / telephone) and fidelity. Fidelity low / medium / high has different meanings for different channel conditions. For the studio speech, high fidelity is used for conversations that take place inside a studio. Usually,

		Fidelity		
Channel		High	Medium	Low
Studio	[8kHz Bandwidth]	Studio	Field	Channel noise
Telephone	[4kHz Bandwidth]	Sounds clear	Noisy	Not intelligible

Table A.1: Coding of channel and fidelity attributes

this is when the anchor person is talking or when a video story is commented by a journalist that is recorded in a studio. Medium fidelity refers to speech that is captured in the field, usually situations where the journalist is making a street interview. Low fidelity refers to situations where there is noise in the transmission channel. In the case of telephone speech, high refers to clear (clean) speech, medium to noisy speech that is still easy to understand though, and low to speech that is difficult to understand.

The speech utterances should not be too long and every speaker inspiration event should be regarded as a potential breakpoint. When a silence inside a speaker turn is less than 0.5 seconds it is not marked at all. When it is between 0.5 and 1.5 seconds, a breakpoint in the middle of the silence is inserted. When the silence is longer than 1.5 seconds, two breakpoints delimiting the silence are inserted. The sections blocks are categorized as reports (news stories), fillers (headlines and short story descriptions) and nontrans (commercials and jingle segments). All jingle segments are identified as such and marked by a noise event tag. When the TV station uses different jingles at the beginning and the end of a show, each jingle gets an additional suffix indicating its begin/end category. Foreign language utterances are marked with language event tags and are not transcribed. We defined a close set of interjections and non-lexemes so as to pursue that the same tags are used for the same words/sounds in all the data.

A.1 LDC focus conditions

For characterizing the acoustic, channel and speaker conditions of BN corpora we adopted the LDC Hub4 Broadcast News focus conditions which are summarized in Table A.2.

Focus	Description
F0	Baseline broadcast speech (clean, planned)
F1	Spontaneous broadcast speech (clean)
F2	Low fidelity speech (narrowband/telephone)
F3	Speech in the presence of background music
F4	Speech in the presence of background noise degraded acoustical conditions (F40 = planned; F41 = Spontaneous)
F5	Non-native speakers (clean, planned)
Fx	All other speech (e.g. spontaneous non-native)

Table A.2: LDC Hub4 focus conditions

Additionally, we split the F4 focus condition into F40 and F41 according to the speech mode (Planned or Spontaneous). In our BN speech corpus almost every set has a very high percentage of speech in the presence of background noise. By spliting the F4 condition we have a clearer notion about the percentage of data with planned or spontaneous speech.

Appendix B

European Portuguese Phoneme set

Table B.1 presents a list of the phonetic symbols used for the European Portuguese.

IPA Symbol	SAM_PA Symbol	Example
Oclusive consonants		
p	p	(p)ai
t	t	(t)ia
k	k	(c)asa
b	b	(b)ar
d	d	(d)ata
g	g	(g)ato
Fricative consonants		
s	s	(s)elo
z	z	a(z)ul
f	f	(f)érias
v	v	(v)aca
ʃ	S	(ch)ave
ʒ	Z	a(g)ir
Liquid consonants		
l	l	(l)ado
ɫ	l~	sa(l)
ʎ	L	fo(lh)a
ɾ	r	ca(r)o
ʀ	R	ca(rr)o
Nasal consonants		
m	m	(m)eta
n	n	(n)eta
ɲ	J	se(nh)a
Semi-vowels		
j	j	pa(i)
w	w	pa(u)
Nasal vowels		
ĩ	i~	p(in)to
õ	o~	p(on)te
ɐ̃	A~	c(an)to
ẽ	e~	d(en)te
ũ	u~	f(un)do
j̃	j~	põ(e)
w̃	w~	mã(o)
Vowels		
ɛ	E	s(e)te
ɔ	O	c(o)rda
u	u	m(u)do
ɨ	@	qu(e)
i	i	f(i)ta
e	e	p(e)ra
a	a	c(a)ra
ɐ	A	c(a)m(a)
o	o	d(ou)

Table B.1: Phonetic symbols used in European Portuguese

Appendix C

Algorithmic Optimizations

Currently Broadcast news speech recognition processing needs large computational resources. It requires many processing steps from the initial recording to the final recognized text string. This involves many complex signal processing calculations. Furthermore, the BN data is normally very complex from the acoustical point of view covering spontaneous speech, hesitations, many different types of background noise, music, channel conditions. Large acoustic models will be required to cope with all the complex data. This emplyes many calculations and will lead to slower performance when running. Large news shows (over one hour) will need many hours to process.

Furthermore, to train such complex acoustic models we will need large training data sets. Normally, this requires several weeks waiting for results (trained acoustic models). We will end up having less flexibility in experimenting with different possibilities. Code optimization becomes a crucial factor.

C.1 Feature extration optimizations

Almost all processing steps in the BN acoustic domain use prior feature extraction. In our ASR system we use PLP [Hermansky et al., 1992], Log-RASTA [Hermansky et al., 1992] and MSG [Kingsbury et al., 1998] feature coefficients. PLP and Log-RASTA share the same code and basically only minor modifications were made. We removed some unnecessary byte swapping and compiled with optimization flags. This yielded a significant decrease in execution time. The MSG features use two different filters in the middle processing stage that in the end produce the two feature vectors we are using, each with 14 features. Normal program execution involved two call to the MSG algorithm, one for each filter type. Doing this we were doing some duplicate calculations since the first complex processing stages of MSG (up to the middle filters) are exactly equal. By embedding the two filters in the code we removed duplicate calculations. Combined with optimization flags for compilation yielded another significant decrease in running time.

Experiments were conducted using a complete news show with 35 minutes from the ALERT-SR.train set. In table C.1 we show the execution times for each feature extraction process before and after the optimization procedures.

Feature extraction process	Before (sec)	After (sec)	Speedup factor
PLP	11	7	1.6
Log-RASTA	11	5	2.2
MSG	23	5	4.6

Table C.1: Feature extraction optimizations test.

We achieved a reduction in execution time superior to 50%. The reduction in MSG was more significative due to the elimination of the repeated calculations.

C.2 MLP code optimizations

The MLPs forward pass computing is the step in the acoustic model calculation that consumes most time because of the very large neural networks which have hundreds of thousand parameters and thousands of non-linear sigmoidal functions. We wanted to optimize the code for both training routines (forward and backward pass) and testing / running forward only pass. Training has more optimisation potential because it has more than double operations due to both passes (forward and backward).

MLP forward and backward passes were recoded in order to became vector-matrix and matrix-matrix multiplication and sum operations. Sigmoidal function was implemented by a pre-computed lookup table with a requested number of entries, normally one thousand. The vector and matrix operations were then implemented by optimized routines called BLAS (Basic Linear Algebra Subprograms) [NETLIB, 2000]. There are several available BLAS packages (open source and commercial). We tested several approaches, namely ATLAS [ATLAS, 2000], intel Math Kernel Library [Intel MKL, 2006] and GOTO BLAS [Goto, 2006]. These BLAS packages are specifically optimized for a certain CPU architecture (Intel Pentium in our case) do sub-block optimization depending on system cache sizes and can take advantage of multiple CPUs.

MLP simulator	Speedup factor
nns, original code	1.0 x
nns, simple BLAS	2.9 x
nns, intel MKL (bunch = 1)	5.3 x
nns, intel MKL (bunch = 32)	19.9 x
nns, goto blas (bunch = 32)	27.8 x

Table C.2: MLP simulator performance optimization.

The final improvement was to modify the simulator architecture to explore the BLAS routines potential. We converted vector-matrix multiplications into matrix-matrix multiplication which have the most optimized routines inside the BLAS packages. This was achieved by processing multiple input patterns (called bunch processing [Bilmes et al., 1997]). Which is a mix between online stochastic mode and batch mode. In bunch mode weights are only updated after each bunch forward, meaning that weights are not updated after every forward of an input pattern. On the one hand less weight updates possibly mean a slower training (potentially more training epochs needed). On the other hand the batch update means a more precise gradient vector update [Bilmes et al., 1997]. Table C.2 shows the speedup factor obtained by successive code optimisations. These tests were obtained in the same PC. The total speedup achieved is quite impressive.

Bibliography

[Almeida, 1996] Almeida, L. (1996). *Handbook of Neural Computation*, chapter Multilayer Perceptrons. IOP Publishing Ltd and Oxford University Press, UK.

[Amaral et al., 2001] Amaral, R., Langlois, T., Meinedo, H., Neto, J., Souto, N., and Trancoso, I. (2001). *The development of a Portuguese version of a media watch system*. In Proceedings EUROSPEECH 2001, Aalborg, Denmark.

[Amaral et al., 2006] Amaral, R., Meinedo, H., Caseiro, D., Trancoso, I., and Neto, J. (2006). *Automatic vs. Manual Topic Segmentation and Indexation in Broadcast News*. In IV Jornadas en Tecnologia del Habla, Saragoza, Spain.

[Amaral et al., 2007] Amaral, R., Meinedo, H., Caseiro, D., Trancoso, I., and Neto, J. (2007). *A Prototype System for Selective Dissemination of Broadcast News in European Portuguese*. *EURASIP journal on Advances in Signal Processing*, 2007, Article ID 37507.

[Amaral and Trancoso, 2003a] Amaral, R. and Trancoso, I. (2003a). *Indexing Broadcast News*. In Proceedings 3rd International Workshop on New Developments in Digital Libraries – NDDL 2003, Angers, France.

[Amaral and Trancoso, 2003b] Amaral, R. and Trancoso, I. (2003b). *Segmentation and Indexation of Broadcast News*. In Proceedings ISCA Workshop on Multilingual Spoken Document Retrieval – MSDR 2003, Hong Kong, China.

[Amaral and Trancoso, 2003c] Amaral, R. and Trancoso, I. (2003c). *Topic Indexing of TV Broadcast News Programs*. In Proceedings 6th International Workshop on Computational Processing of the Portuguese Language – PROPOR 2003, Faro, Portugal.

[Amaral and Trancoso, 2004] Amaral, R. and Trancoso, I. (2004). *Improving the Topic Indexation and Segmentation Modules of a Media Watch System*. In Proceedings ICSLP 2004, Jeju, Korea.

[ATLAS, 2000] ATLAS (2000). *Automatically Tuned Linear Algebra Software (ATLAS)*. http://math-atlas.sourceforge.net.

[Bakis et al., 1998] Bakis, R., Chen, S., Gopalakrishnan, P., Gopinath, R., Maes, S., Polymenakos, L., and Franz, M. (1998). *Transcription of Broadcast News Shows with the IBM Large Vocabulary Speech Recognition System*. In Proceedings DARPA Speech Recognition Workshop 1998.

[Barras et al., 1998] Barras, C., Geoffrois, E., Wu, Z., and Liberman, M. (1998). *Transcriber: a Free Tool for Segmenting, Labeling and Transcribing Speech*. In Proceedings 1st International Conference on Language Resources and Evaluation – LREC 1998.

[Barras et al., 2001] Barras, C., Geoffrois, E., Wu, Z., and Liberman, M. (2001). *Transcriber: development and use of a tool for assisting speech corpora production*. *Speech Communication*, 33(1–2):5–22.

[Barzilay et al., 2000] Barzilay, R., Collins, M., Hirschberg, J., and Whittaker, S. (2000). *The Rules Behind Roles: Identifying Speaker Role in Radio Broadcast*. In Proceedings AAAI 2000, Austin, USA.

[Berger et al., 1996] Berger, A. L., Pietra, S. D., and Pietra, V. J. D. (1996). *A Maximum Entropy Approach to Natural Language Processing*. *Computational Linguistics*, 22(1):39–71.

[Bilmes et al., 1997] Bilmes, J., Asanovic, K., Chin, C., and Demmel, J. (1997). *Using PHiPAC to speed Error Back-Propagation Learning*. In Proceedings ICASSP 1997, Munich, Germany.

[Bourlard and Morgan, 1994] Bourlard, H. and Morgan, N. (1994). *Connectionist Speech Recognition - A Hybrid Approach.* Kluwer Academic Publishers, Massachusetts, EUA.

[Caseiro and Trancoso, 2000] Caseiro, D. and Trancoso, I. (2000). *A Decoder for Finite-State Structured Search Spaces.* In ASR 2000 Workshop, Paris, France.

[Caseiro and Trancoso, 2001a] Caseiro, D. and Trancoso, I. (2001a). *On Integrating the Lexicon with the Language Model.* In Proceedings EUROSPEECH 2001, Aalborg, Denmark.

[Caseiro and Trancoso, 2001b] Caseiro, D. and Trancoso, I. (2001b). *Transducer Composition for "On-the-Fly" Lexicon and Language Model Integration.* In Proceedings ASRU Workshop 2001, Madonna di Campiglio, Trento, Italy.

[Caseiro and Trancoso, 2002] Caseiro, D. and Trancoso, I. (2002). *Using Dynamic WFST Composition for Recognizing Broadcast News.* In Proceedings ICSLP 2002, Denver, USA.

[Caseiro and Trancoso, 2003] Caseiro, D. and Trancoso, I. (2003). *A Tail-Sharing WFST Composition for Large Vocabulary Speech Recognition.* In Proceedings ICASSP 2003, Hong Kong, China.

[Chen and Gopalakrishnan, 1998] Chen, S. and Gopalakrishnan, P. (1998). *Speaker, Environment And Channel Change Detection And Clustering Via The Bayesian Information Criterion.* In Proceedings DARPA Speech Recognition Workshop 1998.

[Delacourt et al., 2000] Delacourt, P., Bonastre, J., Fredouille, C., Merlin, T., and Wellekens, C. (2000). *A Speaker Tracking System Based on Speaker Turn Detection for NIST Evaluation.* In Proceedings ICASSP 2000, Istanbul, Turkey.

[Delacourt and Wellekens, 2000] Delacourt, P. and Wellekens, C. (2000). *DISTBIC: A speaker-based segmentation for audio data indexing*. *Speech Communication*, 32:111–116.

[El-Maleh et al., 2000] El-Maleh, M., Klein, G., and Kabal, P. (2000). *Speech/ music discrimination for multimedia application*. In Proceedings ICASSP 2000, Istanbul, Turkey.

[Ellis and Morgan, 1999] Ellis, D. and Morgan, N. (1999). *Size matters: an Empirical Study of Neural Network Training for Large Vocabulary Continuous Speech Recognition*. In Proceedings ICASSP 1999, USA.

[Gales et al., 2007] Gales, M., Kim, D., Woodland, P., Chan, H., Mrva, D., Sinha, R., and Tranter, S. (2007). *Progress in the CU-HTK Broadcast News Transcription System*. *IEEE Transactions on Audio, Speech and Language Processing*.

[Galliano et al., 2005] Galliano, S., Geoffrois, E., Mostefa, D., Choukri, K., Bonastre, J.-F., and Gravier, G. (2005). *The ESTER Phase II Evaluation Campaign for the Rich Transcription of French Broadcast News*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Garofolo et al., 2000] Garofolo, J., Auzanne, G., and Voorhees, E. (2000). *The TREC Spoken Document Retrieval Track: A Success Story*. In Proceedings of the Recherche d'Informations Assiste par Ordinateur – RIAO 2000.

[Gauvain et al., 1995] Gauvain, J., Lamel, L., and Adda, G. (1995). *Developments in Continuous Speech Dictation using the ARPA WSJ Task*. In Proceedings ICASSP 1995, Detroit, USA.

[Gauvain et al., 1998] Gauvain, J.-L., Lamel, L., and Adda, G. (1998). *Partitioning and Transcription of broadcast news data*. In Proceedings ICASSP 1998, USA.

[Gelbukh et al., 2001] Gelbukh, A., Sidorov, G., and Guzmán-Arenas, A. (2001). *Document Indexing With a Concept Hierarchy*. In Proceedings 1st International Workshop on New Developments in Digital Libraries – NDDL 2001, Setúbal, Portugal.

[Gish et al., 1991] Gish, H., Siu, H., and Rohlicek, R. (1991). *Segregation of speakers for speech recognition and speaker identification*. In Proceedings ICASSP 1991, Toronto, Canada.

[Goto, 2006] Goto, K. (2006). *Goto High performance BLAS*. `http://www.tacc.utexas.edu/~kgoto`.

[Group, 2001] Group, N. S. (2001). *The 2001 Topic Detection and Tracking* (TDT2001) *Task Definition and Evaluation Plan*.

[Hermansky et al., 1992] Hermansky, H., Morgan, N., Baya, A., and Kohn, P. (1992). *RASTA-PLP Speech Analysis Technique*. In Proceedings ICASSP 1992, San Francisco, USA.

[Intel MKL, 2006] Intel MKL (2006). *Intel Math Kernel Library (MKL)*. `http://www.intel.com/cd/software/products/asmo-na/eng/perflib/index.htm`.

[Istrate et al., 2005] Istrate, D., Scheffer, N., Fredouille, C., and Bonastre, J.-F. (2005). *Broadcast News Speaker Tracking for ESTER 2005 Campaign*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Jin et al., 1997] Jin, H., Kubala, F., and Schwartz, R. (1997). *Automatic speaker clustering*. In Proceedings DARPA Speech Recognition Workshop.

[Johnson and Woodland, 1998] Johnson, S. and Woodland, P. (1998). *Speaker clustering using direct maximization of the MLLR adapted likelihood*. In Proceedings ICSLP 1998, Sydney, Australia.

[Kemp et al., 2000] Kemp, T., Schmidt, M., Westphal, M., and Waibel, A. (2000). *Strategies for automatic segmentation of audio data*. In Proceedings ICASSP 2000, Istanbul, Turkey.

[Kingsbury et al., 1998] Kingsbury, B. E., Morgan, N., and Greenberg, S. (1998). *Robust Speech Recognition using the Modulation Spectrogram*. *Speech Comunication*, 25:117–132.

[Lamel et al., 2004] Lamel, L., Gauvain, J., and Canseco-Rodriguez, L. (2004). *Speaker Diarization from Speech Transcripts*. In Proceedings ICSLP 2004, Jeju, Korea.

[Lawrence et al., 1998] Lawrence, S., Burns, I., Back, A., Tsoi, A., and Giles, C. (1998). *Neural Networks: Tricks of the Trade*, volume 1524 of *Lecture Notes in Computer Science*, chapter Neural Network classification and Prior class probabilities, pp. 299–314. Springer Verlag.

[LDC-Hub4, 2000] LDC-Hub4 (2000). *http://www.ldc.upenn.edu/Projects/Corpus_Cookbook/transcription/broadcast_speech/english/index.html*.

[Liu and Kubala, 1999] Liu, D. and Kubala, F. (1999). *Fast speaker change detection for broadcast news transcription and indexing*. In Proceedings EUROSPEECH 1999, Budapest, Hungary.

[Liu and Kubala, 2003] Liu, D. and Kubala, F. (2003). *Online Speaker Clustering*. In Proceedings ICASSP 2003, Hong Kong, China.

[Liu and Kubala, 2005] Liu, D. and Kubala, F. (2005). *Online Speaker adaptation and tracking for real-time speech recognition*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Mahalanobis, 1936] Mahalanobis (1936). *Mahalanobis Distance*. http://en.wikipedia.org/wiki/Mahalanobis_distance.

[Martins et al., 2005] Martins, C., Teixeira, A., and Neto, J. (2005). *Language Models in Automatic Speech Recognition. Revista Electrónica e Telecomunicações, Departamento de Electrónica e Telecomunicações, Universidade de Aveiro*, 4(4).

[Matsoukas et al., 2005] Matsoukas, S., Prasad, R., Laxminarayan, S., Xiang, B., Nguyen, L., and Schwartz, R. (2005). *The 2004 BBN 1xRT Recognition Systems for English Broadcast News and Conversational Telephone Speech*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Meinedo et al., 2003] Meinedo, H., Caseiro, D., Neto, J., and Trancoso, I. (2003). *AUDIMUS.*MEDIA*: A Broadcast News Speech Recognition System for the European Portuguese Language*. In Proceedings 6th International Workshop on Computational Processing of the Portuguese Language – PROPOR 2003, Faro, Portugal.

[Meinedo and Neto, 2000] Meinedo, H. and Neto, J. (2000). *Combination of acoustic models in continuous speech recognition*. In Proceedings ICSLP 2000, Beijing, China.

[Meinedo and Neto, 2003a] Meinedo, H. and Neto, J. (2003a). *Audio segmentation, classification and clustering in a Broadcast News task*. In Proceedings ICASSP 2003, Hong Kong, China.

[Meinedo and Neto, 2003b] Meinedo, H. and Neto, J. (2003b). *Automatic speech annotation and transcription in a Broadcast News task*. In Proceedings ISCA Workshop on Multilingual Spoken Document Retrieval – MSDR 2003, Hong Kong, China.

[Meinedo and Neto, 2004] Meinedo, H. and Neto, J. (2004). *Detection of Acoustic Patterns in Broadcast News using Neural Networks*. In Proceedings Acoustics European Symposium – Acustica 2004, Guimarães, Portugal.

[Meinedo and Neto, 2005] Meinedo, H. and Neto, J. (2005). *A Stream-based Audio Segmentation, Classification and Clustering Pre-processing System for Broadcast News using ANN Models*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Meinedo et al., 2001] Meinedo, H., Souto, N., and Neto, J. (2001). *Speech Recognition of Broadcast News for the European Portuguese Language*. In Proceedings ASRU Workshop 2001, Madonna di Campiglio, Trento, Italy.

[Mermelstein, 1976] Mermelstein, P. (1976). *Pattern Recognition and Artificial Intelligence*, chapter Distance measures for Speech Recognition, Psychological and Instrumental, pp. 374–388. Academic, New York.

[Moh et al., 2003] Moh, Y., Nguyen, P., and Junqua, J.-C. (2003). *Toward domain independent clustering*. In Proceedings ICASSP 2003, Hong Kong, China.

[Moraru et al., 2005] Moraru, D., Ben, M., and Gravier, G. (2005). *Experiments on Speaker Tracking and Segmentation in Radio Broadcast News*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Moraru et al., 2004] Moraru, D., Meignier, S., Fredouille, C., Besacier, L., , and Bonastre, J. (2004). *The ELISA consortium approaches in Broadcast News speaker segmentation during the NIST 2003 Rich Transcription evaluation*. In Proceedings ICASSP 2004, Philadelphia, USA.

[NETLIB, 2000] NETLIB (2000). *BLAS optimization routines archive*. `http://www.netlib.org/blas/`.

[Neto, 1998] Neto, J. (1998). *Reconhecimento de fala contíua com aplicações de técnicas de adaptação ao orador*. Tese de Doutoramento, IST, Lisboa, Portugal.

[Neto et al., 1996] Neto, J., Martins, C., and Almeida, L. (1996). *An Incremental Speaker-Adaptation Technique for Hybrid HMM-MLP Recognizer*. In Proceedings ICSLP 1996, Philadelphia, USA.

[Neto et al., 1997a] Neto, J., Martins, C., and Almeida, L. (1997a). *The Development of a Speaker Independent Continuous Speech recognizer for Portuguese*. In Proceedings EUROSPEECH 1997, Rhodes, Greece.

[Neto et al., 1998] Neto, J., Martins, C., and Almeida, L. (1998). *A large vocabulary continuous speech recognition hybrid system for the Portuguese language*. In Proceedings ICSLP 1998, Sydney, Australia.

[Neto et al., 1997b] Neto, J., Martins, C., Meinedo, H., and Almeida, L. (1997b). *The Design of a Large Vocabulary Speech Corpus for Portuguese*. In Proceedings EUROSPEECH 1997, Rhodes, Greece.

[Neto et al., 2003a] Neto, J., Meinedo, H., Amaral, R., and Trancoso, I. (2003a). *The development of an automatic system for selective dissemination of multimedia information*. In Proceedings Third International Workshop on Content-Based Multimedia Indexing – CBMI 2003, Rennes, France.

[Neto et al., 2003b] Neto, J., Meinedo, H., Amaral, R., and Trancoso, I. (2003b). *A System for Selective Dissemination of Multimedia Information*. In Proceedings ISCA Workshop on Multilingual Spoken Document Retrieval – MSDR 2003, Hong Kong, China.

[Nguyen et al., 2005] Nguyen, L., Xiang, B., Afify, M., Abdou, S., Matsoukas, S., Schwartz, R., and Makhoul, J. (2005). *The BBN RT04 English Broadcast News Transcription System*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[NIST, 2000] NIST (2000). *Speech Recognition Scoring Toolkit (SCTK)*. `http://www.nist.gov/speech/tools/`.

[NIST, 2004] NIST (2004). *Fall 2004 Rich Transcription (RT-04F) evaluation plan*.

[NIST RT, 2003] NIST RT (2003). *NIST Rich Text evaluation*. `http://www.nist.gov/speech/tests/rt/`.

[Paul and Baker, 1992] Paul, D. and Baker, J. (1992). *The Design for the Wall Street Journal-based CSR Corpus*. In Proceedings ICSLP 1992, Alberta, Canada.

[Ramabhadran et al., 2003] Ramabhadran, B., Huang, J., Chaudhari, U., Iyengar, G., and Nock, H. (2003). *Impact of audio segmentation and segment clustering on automated transcription accuracy of large spoken archives*. In Proceedings EUROSPEECH 2003, Geneve, Switzerland.

[Renals and Hochberg, 1995a] Renals, S. and Hochberg, M. (1995a). *Decoder Technology for Connectionist Large Vocabulary Speech Recognition*. Relatório Técnico CUED/F-INFENG/TR.186, Cambridge University Engineering Department, Cambridge, England.

[Renals and Hochberg, 1995b] Renals, S. and Hochberg, M. (1995b). *Efficient Search Using Posterior Phone Probability Estimates*. In Proceedings ICASSP 1995, Detroit, USA.

[Renals and Hochberg, 1999] Renals, S. and Hochberg, M. (1999). *Start-synchronous search for large vocabulary continuous speech recognition*. *IEEE Transactions on Speech and Audio Processing*, 5(7):542–553.

[Sheirer and Slaney, 1997] Sheirer, E. and Slaney, M. (1997). *Construction and evaluation of a robust multifeature speech/music discriminator*. In Proceedings ICASSP 1997, Munich, Germany.

[Shriberg, 2005] Shriberg, E. (2005). *Spontaneous Speech: How People Really Talk, and Why Engineers Should Care*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Siegler et al., 1997] Siegler, M., Jain, U., Raj, B., and Stern, R. (1997). *Automatic Segmentation, Classification and clustering of Broadcast News*. In Proceedings DARPA Speech Recognition Workshop.

[Sinha and et al., 2005] Sinha, R. and et al. (2005). *The Cambridge University March 2005 Speaker Diarisation System*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Souto et al., 2002] Souto, N., Meinedo, H., and Neto, J. (2002). *Building Language Models for Continuous Speech Recognition Systems*. In Proceedings Portugal for Natural Language Processing – PorTAL 2002, Faro, Portugal.

[SPECOM, 2002] SPECOM (2002). *Articles in Issues 1-2*. *Speech Communication 2002*, 37:1–159.

[Trancoso et al., 2003] Trancoso, I., Neto, J., Meinedo, H., and Amaral, R. (2003). *Evaluation of an alert system for selective dissemination of broadcast news*. In Proceedings EUROSPEECH 2003, Geneve, Switzerland.

[Trancoso et al., 2004] Trancoso, I., Neto, J., Meinedo, H., Amaral, R., and Caseiro, D. (2004). *An Acoustic Driven Media Watch System*. In Proceedings Acoustics European Symposium – Acustica 2004, Guimarães, Portugal.

[Tranter, 2004] Tranter, S. (2004). *Who really spoke when ? - Finding speaker turns and identities in audio*. In Proceedings ICASSP 2004, Philadelphia, USA.

[Tranter and Reynolds, 2004] Tranter, S. and Reynolds, D. (2004). *Speaker Diarization for broadcast news*. In Proceedings Odyssey Speaker and Language Recognition Workshop, Toledo, Spain.

[Tranter and Reynolds, 2006] Tranter, S. and Reynolds, D. (2006). *An Overview of Automatic Speaker Diarization Systems*. *IEEE Transactions on Audio, Speech and Language Processing*, 14(5):1557–1565.

[Tritschler and Gopinath, 1999] Tritschler, A. and Gopinath, R. (1999). *Improved speaker segmentation and segments clustering using the Bayesian information criterion*. In Proceedings EUROSPEECH 1999, Budapest, Hungary.

[Vandecatseye and Martens, 2003] Vandecatseye, A. and Martens, J. (2003). *A fast, accurate and stream-based speaker segmentation and clustering algorithm*. In Proceedings EUROSPEECH 2003, Geneve, Switzerland.

[Vandecatseye et al., 2004] Vandecatseye, A., Martens, J., Neto, J., Meinedo, H., Mateo, C., Dieguez, J., Mihelic, F., Zibert, J., Nouza, J., David, P., Pleva, M., Cizmar, A., Papageorgiou, H., and Alexandris, C. (2004). *The COST278 pan-European Broadcast News Database*. In Proceedings International Conference on Language Resources and Evaluation – LREC 2004, Lisbon, Portugal.

[Wessel and Ney, 2001] Wessel, F. and Ney, H. (2001). *Unsupervised training of acoustic models for large vocabulary continuous speech recognition*. In Proceedings ASRU Workshop 2001, Madonna di Campiglio, Trento, Italy.

[Wessel and Ney, 2005] Wessel, F. and Ney, H. (2005). *Unsupervised training of acoustic models for large vocabulary continuous speech recognition*. *IEEE Transactions on Speech and Audio Processing*.

[Williams, 1999] Williams, D. (1999). *Knowing what you don't know: roles for confidence measures in automatic speech recognition*. Tese de Doutoramento, Univ. Sheffield, UK.

[Williams and Ellis, 1999] Williams, G. and Ellis, D. (1999). *Speech/music discrimination based on posterior probability features*. In Proceedings EUROSPEECH 1999, Budapest, Hungary.

[Woodland et al., 1997] Woodland, P., Gales, M., Pye, D., and Young, S. (1997). *Broadcast News Transcription Using HTK*. In Proceedings ICASSP 1997, Munich, Germany.

[Zdansky et al., 2004] Zdansky, J., David, P., and Nouza, J. (2004). *An Improved Preprocessor for the Automatic Transcription of Broadcast News Audio Stream*. In Proceedings ICSLP 2004, Jeju, Korea.

[Zhou and Hansen, 2000] Zhou, B. and Hansen, J. (2000). *Unsupervised audio stream segmentation and clustering via the Bayesian Information Criterion*. In Proceedings ICSLP 2000, Beijing, China.

[Zhu et al., 2005a] Zhu, Q., Chen, B., Grezl, F., and Morgan, N. (2005a). *Improved MLP Structures for Data-Driven Feature Extraction for ASR*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Zhu et al., 2005b] Zhu, Q., Stolcke, A., Chen, B., and Morgan, N. (2005b). *Using MLP Features in SRI's Conversational Speech Recognition System*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Zhu and et al., 2005] Zhu, X. and et al. (2005). *Combining speaker identification and BIC for speaker diarization*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Zibert et al., 2005] Zibert, J., Mihelic, F., Martens, J., Meinedo, H., Neto, J., Docio, L., Garcia-Mateo, C., David, P., Nouza, J., Pleva, M., Cizmar, A., Zgank, A., Kacic, Z., Teleki, C., and Vicsi, K. (2005). *The COST278 Broadcast News Segmentation and Speaker Clustering Evaluation - Overview, Methodology, Systems, Results*. In Proceedings INTERSPEECH 2005, Lisbon, Portugal.

[Zibert et al., 2006] Zibert, J., Pavesic, N., and Mihelic, F. (2006). *Speech/Non-Speech Segmentation Based on Phoneme Recognition Features*. *EURASIP Journal on Applied Signal Processing*, 2006, Article ID 90495.