A SYSTEM FOR SELECTIVE DISSEMINATION OF MULTIMEDIA INFORMATION RESULTING FROM THE ALERT PROJECT

João P. Neto [1] **Hugo Meinedo** [1] **Rui Amaral** [2] **Isabel Trancoso** [1]

[1] Instituto Superior Técnico / INESC ID Lisboa
[2] Instituto Politécnico de Setúbal / INESC ID Lisboa

L^{2}F - Spoken Language Systems Laboratory
INESC ID Lisboa, R. Alves Redol, 9, 1000-029 Lisboa, Portugal

{Joao.Neto, Hugo.Meinedo, Rui.Amaral, Isabel.Trancoso}@l2f.inesc-id.pt
http://l2f.inesc-id.pt

Abstract

The media monitoring activity is undergoing a large expansion as a consequence of the different emerging media sources. This is pushing the development of automatic systems for selective dissemination of multimedia information. In this paper we present the development of a prototype system able to scan multimedia data, specifically TV broadcasts, and to generate alert messages to users about the relevant information to them. The system makes use of advanced processing technologies for content-based indexing of multimedia information. We use large vocabulary speech recognition system, associated with audio segmentation, and automatic topic indexing and segmentation, to generate category information as semantic markup of multimedia data. The system service is based on a web interface design offering new views of these marked documents and providing useful end-user services based on the content multimedia exploitation.

1. Introduction

Selective dissemination of information (SDI) is a widely used concept in the scope of written text retrieval from libraries or databases, with large application to some specific activities, as for example, medicine. However, with the rapid expansion of different media sources (newspapers, radio, television, internet), it is difficult to keep track of overall strategic information both in business and governmental agencies, private/public, ... The last few years have shown a large market demand for watching these sources, showing that media monitoring is a crucial activity emerging as a new and strong business area. However, these activities are largely based on manual processing to discern the meaning of the information and its relevance to the end-user needs. A single human operator has to decide if a story has some interest for one of his clients from a set of thousands. It is clear that, without automatic processing, it is not possible to guarantee access to this information to a common user, in a short period of time and at a reasonable economic value.

To accomplish this task, a project named ALERT (*Alert system for selective dissemination of multimedia information*) [1] was proposed in the scope of the Human Language Technologies Program of the IST European framework. The project (IST-1999-10354) started in March 2000 and ran for 30 months, extended for 2.5 additional months. The team included several institutions from 3 countries: Germany, France and Portugal. There were research partners - Gerhard-Mercator Universität Duisburg (D) as coordinator, together with LIMSI (F) and INESC ID Lisboa (P); integrator partners - Vecsys (F) and 4VDO (P); and user representative partners - RTP (P), SECODIP (F) and Observer-Argus Media (D).

The main goal was to use advanced processing technologies for content-based indexing and management of multimedia information to empower the user to select and receive only the information required, even when faced with an ever increasing range of heterogeneous sources. We started by developing a prototype able to scan multimedia data, specifically TV broadcasts, that using a large vocabulary speech recognition system, associated with audio and video segmentation, and automatic topic indexing and segmentation, could generate alert messages to users about the relevant information to them. With this we are building an automatic system, which is performing a selective dissemination of multimedia information.

The implementation of the prototype system started by the definition of a common and generic structure for all the ALERT project. Due to the different user requirements, however, this generic system had to be adapted and tailored to each user and language. This paper describes the SSNT (*Sumarização de Serviços Noticiosos Telivisivos*) prototype implementation for the Portuguese language and RTP's specific objectives and needs. In [2,3] the reader may find some system descriptions for the other languages.

Despite defining different media sources as monitoring targets, at this stage we focus only on TV broadcasts, given RTP's main function as the national TV broadcaster. TV monitoring is very demanding since it involves audio and video processing, together with text processing. Moreover, it involves the development of a management structure to store and retrieve a large set of multimedia documents, and the development of a user interface able to deal with multimedia data. All the other media sources, as radio, newspapers and internet, demand only a subset of these features.

In our prototype the processing stage intends to make a multimedia content analysis, generating category information as a kind of semantic markup of these multimedia data. The acquisition and representation of knowledge is based on thesauri classification of data and the resource description is based on an XML representation. The XML provides an important syntactical foundation upon which the most relevant issues of representing relationships and meaning can be built. Also the development of query languages for XML data enables the development of sophisticated filtering mechanisms that take structure into account. In our implementation, we built a digital library based on a database management service including the multimedia data and the acquired metadata information extracted from the XML data, in conjunction with the interest profiles of the end-users. In order to make the information matching a set of search schemes and triggering services were built. The system service is based on a web interface design offering new views of these marked documents, and providing useful end-user new services based on the content multimedia exploitation.

In this paper we will make an overall presentation of the prototype implementation with a major focus on the system service block. More details of the processing methods are presented in [4,5]. We will start by a description of the Broadcast News (BN) corpus collected to the training and evaluation of the different processing methods. In section 3 we present the system structure followed by the description of the processing block, section 4, and the service block, section 5. Section 6 summarizes the main achievements of this work.

2. Broadcast News Corpus

To support the research and developments associated with this task, it was necessary to collect a representative Portuguese BN corpus both in terms of amount, characteristics and diversity.

We started by defining the type of programs to monitor, in close collaboration with RTP, who 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 news program was selected as the prime target.

This corpus is used for the development of two main modules: the BN speech recognition module and the topic segmentation and indexing module. In that sense we divided our corpus in two parts: the Speech Recognition Corpus and the Topic Detection Corpus. Since each one serves different objectives, each will have different features.

Prior to the collection of these corpus we started with a relative small Pilot Corpus of approximately 5h, including both audio and video, which was used to discuss and setup the collection process, and the most appropriate kind of programs to collect.

The Speech Recognition Corpus was collected next, from November 2000 to January 2001, including 122 programs of different types and schedules and amounting to 76h of audio data. The main goal of this corpus was the training of the acoustic models and the adaptation of the language models used in the large vocabulary speech recognition component of our system.

The last part of our collection effort was the Topic Detection Corpus, which was collected from March through October 2001, containing data related to 133 TV broadcast of the 8 o'clock evening news program. The purpose of this corpus was to have a broader coverage of topics and associated topic classification for training our topic indexation module.

RTP as data provider was responsible for collecting the data in its facilities. The transcription process was jointly done by RTP and INESC ID Lisboa, and made using the Transcriber tool, following the LDC Hub4 (Broadcast Speech) transcription conventions. All the audio data was first automatically transcribed. The orthographic transcriptions of the Pilot Corpus and the Speech Recognition Corpus were both manually verified. For the Topic Detection Corpus, we only have the automatic orthographic transcriptions and the manual segmentation and indexation of the stories made by the RTP staff in charge of the daily program indexing. INESC ID Lisboa was responsible for the training of annotators, for the validation process and for packing the data.

The capture process ran on a PC Windows station equipped with the "MPEG Movie Maker Plus" board (from Optibase) used for MPEG-1 video and audio encoding. Each TV program signal was tuned from the cable network through a stereo video recording whose output was connected directly to the board. We used the SDK Toolkit of the board to build a small application to implement the capture process. When appropriate, this application generated as output only the audio stream in a .wav format.

3. System structure

The system was developed based on a structure of three main blocks: a **CAPTURE** block, responsible for the capture of each of the programs defined to be monitored, a **PROCESSING** block, responsible for the generation of all the relevant markup information for each program, and a **SERVICE** block, responsible for the user and database management interface. A simple scheme of semaphores is used to control the overall process.

In the CAPTURE block we have access to a list of programs to monitor and their time schedule (expected starting and ending time). This time information is the input to a capture program that through a direct access to a TV cable network starts the recording of the specified program at the defined time. This is done using the above mentioned board. This capture program generates an MPEG-1 video and audio encoding file, with the audio at 44.1KHz, 16 bits/sample and stereo. When the recording process is finished, an MPEG-1 file is generated, together with the corresponding semaphore signal that will initialize the next block.

In the PROCESSING block, which will be described in more detail in the next section, the audio stream extracted from the MPEG file is processed through several stages that successively segment, transcribe and index it, compiling the resulting information into an XML file.

In the SERVICE block, we deal with the user interface, through a set of web pages, and database management of user profiles and programs. Each time a new program is processed, an XML file is generated and the database is updated. The matching between the new program and the user profiles generates a list of alerts which are sent to the users through an e-mail service.

The present implementation of the system is focused on demonstrating the usage and features of this system for the 8 o'clock evening news. RTP, the Portuguese user, is interested on indexing every 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. This was in use at RTP in their manual indexing process. This thesaurus follows rules which are generally adopted within EBU (European Broadcast Union).

This thesaurus is an hierarchical structure that covers all possible topics. There are 22 thematic areas in the first level. Each thematic area is subdivided into (up to) 9 lower levels [5]. In our system, we implemented only 3 levels, which are enough to represent the user profile information. Furthermore, 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 indexing. This structure, complemented with geographic (places) and onomastic (names of persons and organizations) descriptors, makes our topic definition.

RTP runs daily a manual process for story segmentation and indexing of the same program. We expect to profit from this manual processing, by comparing the automatic and the manual approaches resulting in a complete evaluation process.

This process is running daily since May 2002 with success. We implemented it at RTP based on a network of 3 machines. These are ordinary PCs running Windows 2000 and Linux. Both the Capture and the Service machines are Windows based and the Processing machine is Linux based. After monitoring one program, all the machines are unloaded with the possibility to manage other programs. The main restriction derives from the Capture machine due to the capture process that is very resource consuming, and supports only a single board. As a consequence, only one program can be monitored at a time unless we add more machines.

4. PROCESSING block

Figure 1 shows a functional diagram of the PROCESSING block.

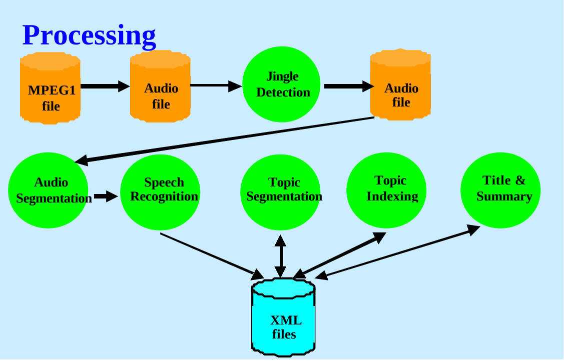


Figure 1. Functional diagram of the Processing block.

This block is based on successive processing stages that transform the MPEG-1 file generated by the CAPTURE block into an XML file containing the orthographic transcription and metadata associated with the audio data.

The first stage extracts the audio file from the MPEG-1 stream downsampling it to 16kHz, disregarding, for the time being, any information that could be derived from the video stream. In the near future, we plan to integrate some image processing techniques to help on different stages of our processing, namely in terms of segmentation [2].

The resulting file is then processed by a ***Jingle Detector***, where we select the program's precise start and ending time, and cut the commercial breaks. The output of this block is an MPEG-1 file, an audio file and the time slots corresponding to the relevant parts of the program. The detector is based on the jingles of the start and ending of the program, and the ones that signal commercials.

The new audio file containing only the relevant parts of the program is then fed through an ***Audio Segmentation*** module [6,7]. This module is used to select only the relevant information and to generate a set of acoustic cues to speech recognition, story segmentation and topic indexing systems. This results in the segmentation of audio into homogeneous regions according to background conditions, speaker gender and special speaker id (anchors). The acoustic segmentation algorithm tries to detect changes in the acoustic conditions and marks those time instants as segment boundaries. Each homogeneous audio segment is then passed through a classification stage in order to tag non-speech segments. All audio segments go through a second classification stage where they are classified according to background status. Segments that were marked as containing speech are also classified according to gender and are subdivided into sentences by an endpoint detector. All labeled speech segments are clustered by gender in order to produce homogeneous clusters according to speaker and background conditions. In the last stage, an anchor detection is done, attempting to identify those speaker clusters that were produced by one of a set of pre-defined news anchors. This segmentation can provide useful information such as division into speaker turns and speaker identities, allowing for automatic indexing and retrieval of all occurrences of a particular speaker. If we group together all segments produced by the same speaker, or by all speakers of the same gender, we can perform an automatic online adaptation of the speech recognition acoustic models to improve the overall system performance.

Each transcribable segment is then processed by the ***Speech Recognition*** module [4]. This module is based on AUDIMUS, a hybrid speech recognition system that combines the temporal modelling capabilities of Hidden Markov Models (HMMs) with the pattern discriminative classification capabilities of multilayer perceptrons (MLPs). 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. Currently our vocabulary is limited to around 60k words associated to a multi-pronunciation lexicon. We use an interpolated 4-gram language model combining a model created from newspaper texts with a model created from BN transcriptions. AUDIMUS presently uses a dynamic decoder that builds the search space as the composition of three Weighted Finite-State Transducers (WFSTs), the HMM/MLP topology transducer, the lexicon transducer and the language model transducer [8]. In [4] you will find evaluation results for this module.

At the end of this module, an XML file is generated containing the audio segments, together with the corresponding markups, and the text transcript of each segment containing speech.

The ***Topic Segmentation*** module [5] processes the XML file and groups segments to define a complete and homogeneous story. Our segmentation algorithm is based on a simple heuristic that from the knowledge of the transcript segments belonging to the anchor defines potential story boundaries in every transition “non-anchor transcript segment / anchor transcript segment”. We refine this heuristic trying to eliminate stories that are too short (containing less than 3 spoken transcript segments), preventing errors on the anchor detection process. This new information of segmentation into stories is added to the XML file.

For each story the ***Topic Indexing*** module [5] generates a classification, according to the hierarchically organized thematic thesaurus, about the contents of the story. The story indexation is performed in two steps. First by detecting the most probable story topic, using the automatically transcribed text for each story. The decoder is based on a HMM methodology and the search for the best hypothesis is accomplished with the Viterbi algorithm. The topology used to model each of the 22 thematic domains is single-state HMMs with self-loops, transition probabilities, and bigram language models. For each of the 22 domains, a smoothed bigram model was built with an absolute discount strategy and a cutoff of 8, meaning that bigrams occurring 8 or fewer times are discarded. The referred models built from the training corpus, give the state observation probabilities. The statistics for each domain were computed from automatically transcribed stories with manually placed boundaries. The corresponding text was post-processed in order to remove all function words and lemmatising the remaining ones. Smoothed bigram statistics were then extracted from this processed corpus. In the second step, we find for the detected domain all the first and second level descriptors that are relevant for the indexation of the story. To accomplish that, 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. 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. The decision to restrict indexation to the second and third node levels was made taking into account the ALERT project goals and the data sparseness at the thesaurus lower levels. This processing is complemented with onomastic and geographic information extracted from the story. All this information is added to the XML file.

Finally a module for generating a ***Title and Summary*** is applied to each story. Since we will deliver to the user a set of stories about the same topic, we would like to have a mechanism to give a close idea about the contents of each story besides the topic indexing. With this goal in mind, we have been working on the extraction of relevant information from the output of the automatic speech recognition module, in order to generate a summary and a title. The most frequent algorithms for summarization are based on sentence or word sequences extraction. When we looked into the structure of the programs that we are monitoring, we observed that the role of the newscaster is to give a brief introduction to the story. Also, his /her first sentence intends to draw the public’s attention to the subject of the story. In our first implementation, we use this knowledge to generate a title based on the first sentence of the newscaster in each story, and the first few sentences to generate its summary. The final result was satisfactory in spite of being completely dependent on the story segmentation process. We are still working on improvements to this method based on relevant word sequences extraction.

At the end of this PROCESSING block, we generate an XML file, according to a DTD specification, containing all the relevant information that we were able to extract.

5. SERVICE block

This block is responsible for the service implementation, in terms of user interface, management of user profiles, loading of the relevant information from the XML file generated in the PROCESSING block into a database structure, and sending alert messages to users resulting from the matching between the program information and their profiles. The database maintains information about the users and about the programs that were processed.

On the user interface there is the possibility to sign up the service, which enables the user to receive alerts on future programs, or to search on the current set of programs for a specific topic. When entering for the first time (registration), or when logging into the system, users have access to two different pages. One page is a form for entering personal information; another is a form for choosing the topics that define the user profile (see Figure 2).

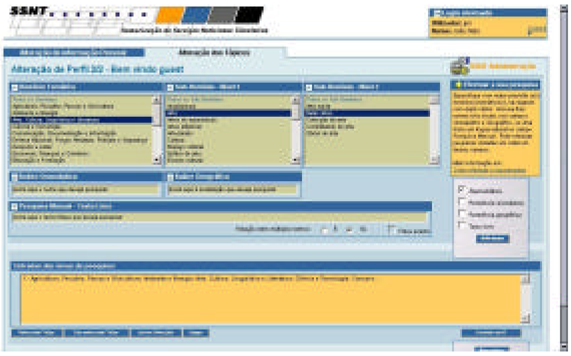


Figure 2. User profile definition.

The profile definition is based on a thematic indexation organized into a hierarchical structure of three levels with additional onomastic and geographical information. This is the same kind of information used in the classification of each story in the PROCESSING block. Additionally, we can further restrict our profile definition to the existence of a free text string. The profile definition results from an AND logic operator on these four kinds of information. This structure allows the selection of more or less specialized topics, depending on the chosen levels and filling of each field. The user can simultaneously select a set of topics, by multiple selection in a specific thematic level, or by entering different individual topics. The combination of these topics can be done through an "AND" or an "OR" boolean operator. After the selection of the topics we can perform a direct search over the current programs in the database or we can logout updating our profile.

When a new XML file is generated, the corresponding semaphore triggers the database loader. The contents of the XML file is read and interpreted to upload the database contents. Figure 3 presents the database UML definition.

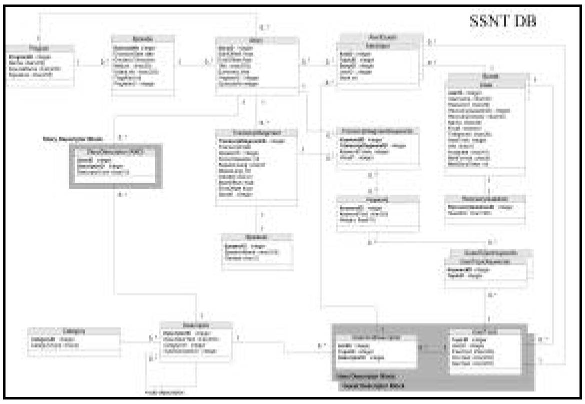
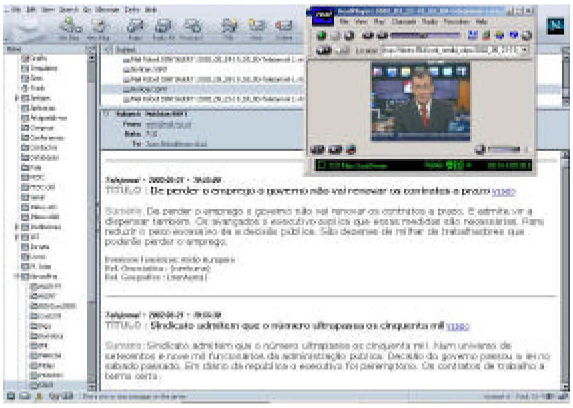


Figure 3. Database UML definition.

The loading of a new program into the database triggers a matching algorithm that searches the new indexed stories, according to the users profile descriptors. This matching algorithm searches for associations between indexing descriptors and relevant words describing each story, and the descriptors and free text composing the user profile. When there is a matching between the two that pair is stored. When the matching process is finished, this triggers a new process that generates the alerts to the registered users. The stories resulting from the matching process are sorted according to the descriptors in the user profile. Each of these stories has an associated title, summary and indexing descriptors, besides a link to the corresponding RealMedia file. With this information we generate an e-mail to the user (see Figure 4). This is the only alert format currently available, but this service is ready to be interconnected with a SMS, MMS or fax server. After sending the e-mail, the same process updates the users table in the database.



TITULO : De perder o emprego o governo não vai renovar os contratos a prazo VIDEO

Sumário: De perder o emprego o governo não vai renovar os contratos a prazo. [illegible]

TITULO : Sindicato admitem que o número ultrapassa os cinquenta mil VIDEO

Sumário: Sindicato admitem que o número ultrapassa os cinquenta mil. [illegible]

Figure 4. Example of an user e-mail.

In order to manage the system, an administration tool was also implemented (see Figure 5). In this tool the administrator has access to all the database contents through a tree structure, in the left part of Figure 5. By choosing a date and a program we get access to all the stories in that program. By choosing a story, we can analyse all the markup information about speakers segments and the full transcription of each of these segments. All the information as topics, onomastic, geographic, title and summary, are also available. In this area we get the RealMedia in SMIL format presenting the transcription as subtitles. This administration area is very useful to monitor the overall process, both from the processing and the service side.

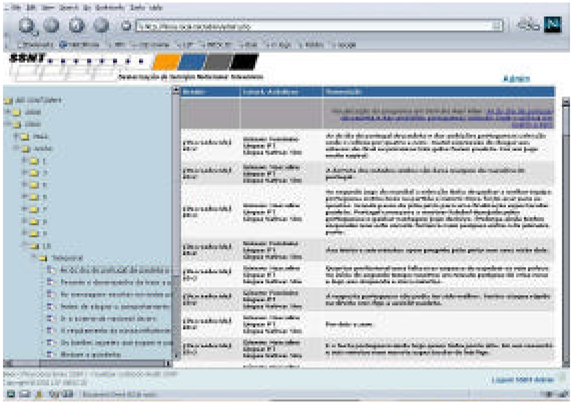


Figure 5. View of the administration area.

In the service implementation, different technologies were used. We decided to make the user interface based on a set of web pages. This will broaden the service to a large universe of potential clients. The http web server was based on Apache, using PHP4 and JavaScipt website scripting language.

In the implementation we used mySQL as the database management system, due to its public domain status and easy implementation. However, since we plan to have a massive user access to the system, other database management structures may be chosen instead.

The Java language was used on the implementation of the topic matching algorithm and on the email generation service. From the client side, the main requirements are a compatible browser and an e-mail reader. Since we deliver extracts of selected programs, a RealMedia streaming server from the server side and a RealMedia player from the user side is also required.

6. Concluding remarks

This new service for selective dissemination of information is currently under a user evaluation phase at RTP and INESC ID Lisboa. From this evaluation we are collecting very interesting feedback over the final users expectations both in terms of service and interface.

The technology that supports this system is under constant evolution, opening the possibility of learning from the daily new data, with positive upgrade of the system.

Presently we are updating the service concept in order to enlarge the number of programs to monitor, to add some radio information programs and to introduce some selected newspapers.

Acknowledgements

This work was partially funded by IST-HLT European program project ALERT and by FCT project POSI/33846/PLP/2000. Hugo Meinedo was also supported by a FCT scholarship SFR/BD/6125/2001. INESC ID Lisboa had support from the POSI Program of the "Quadro Comunitário de Apoio III".
To this work we had the collaboration of several people that we like to thanks: João Sequeira, Hilário Lopes and César Mendes from RTP, António Vaz e Alexandre Mendes from 4VDO and Vítor Pereira and João Gaspar from INESC ID Lisboa.

References

[1] ALERT project web page http://alert.uni-duisburg.de/

[2] Y. Lo and J. L. Gauvain, "The LIMSI Topic Tracking System for TDT 2002", in Proc. DARPA Topic Detection and Tracking Workshop, Gaithersburg, USA, Nov. 2002.

[3] U. Iurgel, S. Werner, A. Kosmala, F. Wallhoff and G. Rigoll, "Audio-Visual analysis of Multimedia documents for automatic topic identification", in Proc. SPPRA 2002, Crete, Greece, June 2002.

[4] H. Meinedo and J. Neto, "Automatic speech annotation and transcription in a Broadcast News task", in Proc. ISCA ITRW on Multilingual Spoken Document Retrieval, Hong Kong, China, April 2003.

[5] R. Amaral and I. Trancoso, "Segmentation and indexation of Broadcast News", in Proc. ISCA ITRW on Multilingual Spoken Document Retrieval, Hong Kong, China, April 2003.

[6] H. Meinedo, N. Souto and J. Neto, "Speech recognition of Broadcast News for the European Portuguese language", in Proc. ASRU 2001, Madonna di Campiglio, Italy, December 2001.

[7] H. Meinedo and J. Neto, "Audio segmentation, classification and clustering in a Broadcast News task", in Proc. ICASSP 2003, Hong-Kong, China, April 2003.

[8] D. Caseiro and I. Trancoso, "A Tail-Sharing WFST Composition for Large Vocabulary Speech Recognition", in Proc. ICASSP 2003, Hong-Kong, China, April 2003.