An Automatic Speech Recognition Solution with Speaker Identification Support

Andi Buzo*, Horia Cucu, Lucian Petrică, Dragoş Burileanu and Corneliu Burileanu
University Politehnica of Bucharest, Speech and Dialogue Research Laboratory, Bucharest, Romania
*Corresponding author (e-mail: andi.buzo@upb.ro)

***Abstract*—Automatic Speech Recognition may suffer in terms of intelligibility if the audio recording contains heterogeneous regions of multiple speakers, music or noise. Diarization is the process of segmenting an audio file into homogeneous regions and when used in conjunction with an Automatic Speech Recognition system, it filters out the non-speech audio regions and significantly improves the intelligibility of the recognition output. In this paper, we present an integrated diarization and transcription solution for the Romanian Language. The solution implements the diarization component as a processing stage in the speech recognizer front end. The integrated system is evaluated in terms of computation efficiency and transcription intelligibility.**

***Keywords*-automatic speech recognition; diarization; speaker recognition.**

I. INTRODUCTION

The purpose of an Automatic Speech Recognition (ASR) system is to produce a text which is as close as possible to the speech content of the audio file being transcribed. The output of such a system is sufficient for applications like audio indexing, command recognition, dictation, etc. There are, however, applications such as broadcast news transcriptions and meeting transcriptions for which the rough output text is not sufficient. Such applications are characterized by many persons speaking in turn, mixed intervals of speech, music, and special effects, or mixed intervals of high quality (e.g., 16 kHz) and telephone audio quality. In these conditions, a rough ASR output text would not be intelligible for audio recordings longer than 20 seconds. Therefore, it is necessary to split the audio recording into homogenous segments before starting the ASR process. A homogenous segment contains speech from the same speaker, with the same channel quality and with the same background conditions (e.g., clean or noisy). The process of dividing the audio recording into homogenous segments and providing labels with information about the segments is called diarization.

The advantages of diarization are multiple. By providing audio segmentation, the ASR output becomes more intelligible, because in most cases the segment's bounds coincide with the end of a sentence. As a stage in the diarization process, the classification of segments can be used for discriminating between speech and non-speech intervals. This problem cannot be solved simply by Voice Activity Detection (VAD) modules which are common in ASR systems. VAD fails in removing music intervals, which generate insertion errors in the ASR output. Filtering speech regions as part of the diarization process eliminates the need of having a separate speech/non-speech detector. Diarization groups the speech segments into clusters according to speaker similarities among the segments. This functionality partially solves the problem of "who spoke when". Finally, the information about the speaker and environment enable the online adaptation of acoustic models to the new conditions (e.g. Maximum Likelihood Linear Regression adaptation). This adaptation can be made online using the information extracted from homogeneous segments. Without diarization, the adaptation is usually made on segments with arbitrary length without guarantee for homogeneity.

Conversely, the diarization raises two important issues. First, diarization can be computationally expensive. Even if the real-time factor of the diarization process is less than 1, it leaves a smaller time frame for the ASR process in order to have an overall real-time application. Second, the diarization process requires a look-ahead of 10 to 20 seconds of speech signal in order to make the segmentation properly. Hence, it introduces a constant delay equal to the look-ahead size. For these reasons, the diarization and ASR are usually used separately.

Most modern diarization methods are based on hierarchical clustering, differing only by the clustering criterion and metrics. The ESTER 2 evaluation campaign [1] revealed important differences among the methods. Methods based on Bayesian Information Criterion (BIC) followed by Cross Likelihood Ratio (CLR) clustering perform well on broadcast news [2, 3]. Systems based on HMM-BIC (Hidden Markov Models - BIC) [4] or T-test distance [5] obtain better results with meeting recordings, while methods based on E-HMM [6] obtain better results on telephone conversation recordings.

In this paper, we propose a solution to the integration of a diarization system with an ASR system. The diarization component is based on the system developed by the LIUM laboratory [7]. The reasons for choosing this system are its open source policy, extensive documentation and its good performance as shown in the ESTER 2 evaluation. The ASR system is the one developed by Speech and Dialogue (SpeeD) Research Laboratory [8] based on the CMU Sphinx4 [9]. This leads to the first large vocabulary ASR with diarization for the Romanian language. We further improve the system with the possibility of speaker identification, using previously trained acoustic models. We measure the computation time of different diarization stages separately, making possible the optimization of the system in order to fit the application requirements.

The rest of the paper is organized as follows: Section II briefly describes the LIUM diarization system. Section III presents the integration solution of the diarization component with the ASR system. Section IV shows the evaluation results of the approach, while conclusions are shown in Section V.

II. THE LIUM DIARIZATION SYSTEM

The diarization system developed by LIUM is a chain of complex processing units as shown in Fig. 1. The feature extraction component is the same used by the CMU Sphinx 4 ASR system, which facilitates the integration of the two systems. The speech features used are the classical Mel Frequency Cepstral Coefficients (MFCCs). The 12 MFCC are completed with the energy and the first and second order derivatives. Depending on the capability of each feature to discriminate environments, genders, speakers, etc., some of them are used only in some stages.

The basic unit in diarization is the segment. The segment is a homogeneous speech interval with only one speaker, the same environment conditions and the same speech quality (telephone, studio). The LIUM system starts with a BIC segmentation, which is based on the speech similarity and uniformity over a speech interval by using a generalized likelihood ratio as metric [7]. A second pass is used for merging similar consecutive segments. The output of this step is a list of homogenous segments. The BIC clustering stage takes this list of segments and tries to group similar segments into clusters. During a speech recording, speakers can talk in turns. The purpose is to group segments that may belong to the same speaker. The metric used at this stage is the same as the metric used in the previous step.

The clusters resulted from the BIC Clustering stage can be used for training Gaussian Mixture Models (GMMs), which are meant to represent each speaker in the recording. The recording is passed to a Viterbi decoding with the trained GMMs. The decoding will classify each speech vector into one of the classes (clusters) producing a new segmentation. Viterbi segmentation is expected to be more accurate than BIC segmentation because the former benefits from global information, while the latter makes decisions based on local information. The output of this stage is a new set of segment clusters.

The next processing stage is the speech/non-speech filtering. This process is a Viterbi decoding using pre-trained GMMs provided by LIUM. The GMMs are one-state HMMs with 8 Gaussian components representing 8 different conditions: 2 models of silence (wide and narrow band), 3 models of speech (clean, over noise and over music), 1 model of narrow band speech, 1 model of jingle and 1 model of music. The speech filtering is very important for the ASR process. A music interval entering the ASR system would result into "garbage text" which is seen as insertion error by the ASR user. This step cannot be done at the beginning of the diarization process because the segments it produces are very fragmented. The filtering of non-speech segments at an early stage would make the BIC segmentation inefficient. Therefore the segmentation produced at this point is combined with the one from the previous steps and the output is a new cluster set where longer non-speech segments are removed. The new segments are labeled with one of the 8 environment conditions.

Gender Detection is similar to Speech/Non-speech Filtering, with a few differences. LIUM provides GMMs for males and females in two different conditions: wide band and narrow band signals. Each speech segment is classified to one of the four GMMs. The segment's boundaries remain the same, but now they are labeled with the proper gender and speech quality (narrow/wide band) information.

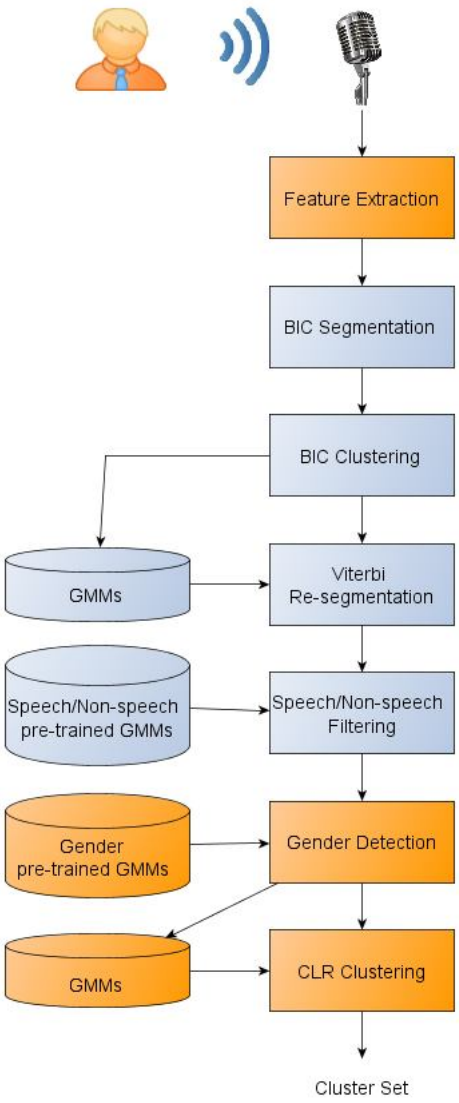

Figure 1. The block diagram of LIUM diarization system

The final processing stage is the CLR Clustering. Its aim is to merge clusters that belong to the same speaker. No feature normalization is used before this stage because it would eliminate the information about the environment. At this stage, the clusters are already labeled with the environment information and feature normalization is made in order to reduce the intra-speaker variation. A Universal Background Model (UBM) is built using the GMMs from the Gender Detection stage. The UBM is adapted with the speech from each cluster in order to obtain a GMM for each speaker. The GMMs are compared two-by-two and the pair that has a similarity over a threshold is merged into one cluster. The similarity metric used is the Cross Entropy [7]. The output of this process is a cluster set of segments with timestamps and labels with environment, gender and speech quality information. Clusters are given speaker IDs that are valid only within the recording being diarized.

III. INTEGRATION OF DIARIZATION WITHIN THE ASR SYSTEM

The CMU Sphinx 4 ASR system has a very flexible front end with the following functionalities: feature extraction, speech/non-speech filtering and feature normalization. Each of these functionalities is implemented as a chain of processing units, which can be easily added or removed from the system using a configuration file. The speech/non-speech filtering is performed by two components: Speech Marker and a data filter. The Speech Marker inserts some data structures called Speech Start and Speech End before and after the speech interval in the feature vector. In the original CMU Sphinx 4, the algorithm for inserting such signals is based on VAD. Subsequently, the Speech Data Filter removes all the feature frames between any two Speech End and Speech Start signals [9].

The integration of the diarization system is shown in Fig. 2. The blocks in green are developed by the SpeeD laboratory. Given the functionalities of diarization, we have decided to integrate the diarization system within a Speech Marker component. The Modified LIUM Diarization contains only the components marked with light blue in Fig. 1. The Feature Extraction is removed because it is performed by ASR system.

Gender Detection and CLR Clustering are removed because we have developed and introduced a Speaker Recognition component. Speaker Recognition makes gender detection and CLR Clustering useless. The clusters produced by the Modified LIUM component are passed to the Speaker Recognition component. The GMMs are trained beforehand with real speakers. They are one-state HMMs with 8 Gaussian components. Different GMMs can be used in different ASR tasks. At this stage the segment bounds are not changed, they are only labeled with the speaker ID. The information from clusters is then used for inserting Speech Start and Speech End signals. The diarization (speaker ID, environment, bandwidth) information is written in the Speech Start signal. The Speech/non-speech Filtering component will remove speech intervals marked as silence, music or jingle.

Some further modifications in the ASR system were required so that the decoder could be aware of the diarization information in the Speech Start signals. This information is passed to the Result data structure which contains the text transcription. Finally, based on the diarization information, the text is formatted in order to increase its intelligibility. Details of such formatting are given in Section IV.

IV. EVALUATION OF THE SYSTEM

The ASR used is the one developed by the SpeeD Research Laboratory for the Romanian language [8]. It uses an acoustic model trained with 54 hours of speech and represented by a HMM pool with 4,000 senones and 16 Gaussian components. The acoustic features used are the MFCC with the energy and the first and second order derivatives. The language model has 64,000 distinct words and is trained with 169 million words.

First, we evaluate the computational efficiency of the integrated system. The results are presented in Table I. The metric used is the real-time (RT) factor. It measures the processing time in seconds for 1 second of audio signal. The database used for evaluation contains 3 hours of audio. The experiments are run on an Intel Xeon E5-2600 series system

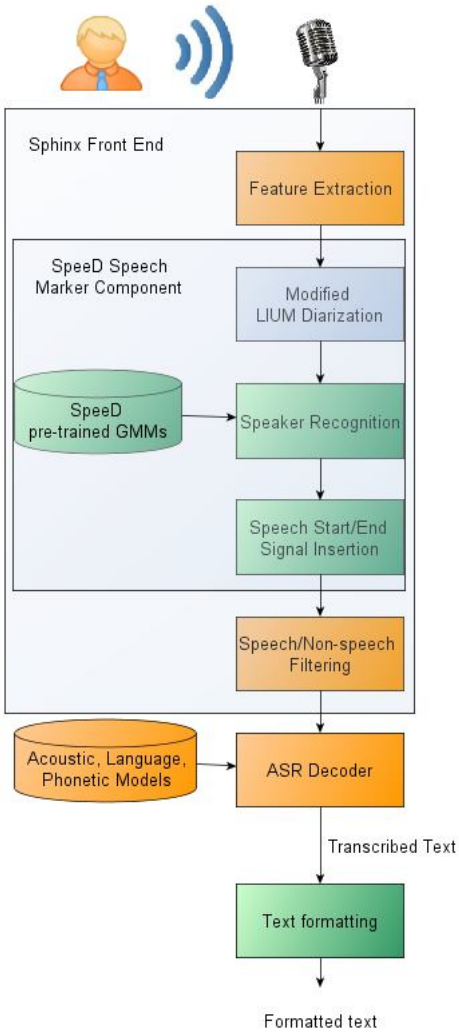

Figure 2. Integration of the diarization component into the ASR system

with 48 GB of RAM. Only one core (2.4 GHz) of this multi-core processor is used. The LIUM diarization has a RT factor of 0.76 and the most time consuming process is CLR Clustering. In the SpeeD solution, the CLR Clustering is replaced by the Speaker Recognition component. Note that the duration of the speaker recognition stage depends on the number of speakers the system can recognize. In this case, we used only 10 GMMs (for 10 speakers). The total RT factor for the SpeeD solution (diarization + ASR) is slightly above real-time. This means that with few optimizations the system can be used in real-time applications such as live radio transcribing. Diarization also helps in eliminating non-speech regions. Such segments are not passed to the ASR decoder, hence reducing the processing time. Obviously, the amount of the processing time reduction depends on the amount of non-speech regions in the audio stream/recording.

Second, the evaluation is made with regard to the transcription intelligibility. Table II shows a comparison between the ASR output without diarization and ASR output with diarization. In the first case the output is just a long text line. In the second case, the output is formatted after a few

rules. The speaker name is added at the beginning of each segment. Consecutive segments belonging to the same speaker are kept in the same line (no new line separation is added between them). The reason for this is to have a similar format with a text dialogue where speakers' quotes are separated in paragraphs. The first word of each segment is capitalized even though the bound of a segment does not always coincide with the beginning of a sentence (it does in more than 70% of the cases). Because in ASR the performance is measured by the number of interventions that a human corrector must make on the output text in order to bring it to the desired form, we decided to add a period after each segment, hence minimizing the number of interventions. Note that the audio used in this example does not contain any music/noise segments, which would have been transcribed into irrelevant words.

Table I
COMPUTING TIME

Diarization Stages	Diarization (x RT)	SpeeD Solution ASR + Diarization (x RT)
Feature extraction	0.01	0.01
BIC Segmentation	0.02	0.02
BIC Clustering	0.02	0.02
Viterbi Decoding	0.04	0.04
Speech Filtering	0.06	0.06
Gender Detection	0.04	n/a
CLR Clustering	0.57	n/a
Speaker recognition	n/a	0.05
ASR	n/a	0.85
Total	**0.76**	**1.05**

Table III
COMPARISON OF ASR OUTPUT WITH/WITHOUT DIARIZATION

ASR output without diarization
salut ce faci bine de unde vii am fost până la universitate trebuia să compar carte am prins un trafic infernal noroc că am ajuns la timp așa e în timpul săptămâni eu de obicei iau metrou desi şi ăla întârzie
ASR output with diarization
Horia: Salut ce faci. Andi: Bine de unde vii. Horia: Am fost până la universitate trebuia să compar carte. Am prins un trafic infernal. Noroc că am ajuns la timp. Andi: Așa e în timpul săptămâni eu de obicei iau metrou desi şi ăla întârzie.
Ideal ASR output
Horia: Salut! Ce faci? Andi: Bine. De unde vii? Horia: Am fost până la universitate. Trebuia să cumpăr o carte. Am prins un trafic infernal. Noroc că am ajuns la timp! Andi: Așa e în timpul săptămânii. Eu de obicei iau metroul, desi şi ăla întârzie.

Overall, the diarization significantly improves the intelligibility of the ASR output. Even though its accuracy is not 100%, the segmentation it provides is better than an arbitrary segmentation that can be used instead. ASR output intelligibility can also be increased using statistical natural language processing (NLP) methods, which are also capable of restoring punctuation marks. However, these text-based methods cannot use the acoustic information (speaker change, speech pauses, etc.) that are available to the diarization method we proposed. We have recently started to work on such NLP post-processing methods and we intend to combine them with the current diarization-based post-processing method.

V. CONCLUSIONS

Diarization is an important process for some ASR applications such as broadcast news transcriptions or meeting transcriptions, where speakers take turns and music or telephone calls may occur. Diarization removes the non-speech intervals and segments the speech intervals into homogenous ones. It also offers support for speaker recognition. If GMMs are trained in advance with known speakers, the homogeneous segments will be classified and labeled with the proper speaker name.

In this paper, we proved that diarization can be fitted in the front end of our ASR system in a natural way and that the solution can be used in real time applications. To the best of our knowledge this is the first diarization-enabled ASR system for the Romanian language.

ACKNOWLEDGMENT

This study has been partially developed with the financial support of the Romanian-American Foundation. The opinions, findings and conclusions or recommendations expressed herein are those of the author(s) and do not necessarily reflect those of the Romanian-American Foundation.

REFERENCES

[1] S. Galliano, G. Gravier, L. Chaubard, "The ester 2 evaluation campaign for the rich transcription of French radio broadcasts," In Proc. Interspeech, pp. 2583-2586, 2009.

[2] C. Barras, X. Zhu, S. Meignier, and J.L. Gauvain, "Multi-stage speaker diarization of broadcast news," IEEE Transactions on Audio, Speech and Language Processing, vol. 14, no. 5, pp. 1505–1512, 2006.

[3] P. Deleglise, Y. Esteve, S. Meignier, and T. Merlin, "Improvements to the LIUM French ASR system based on CMU Sphinx: what helps to significantly reduce the word error rate?" In Proc. Interspeech pp. 2123-2126, 2009.

[4] J. M. Pardo, X. Anguera, and C. Wooters, "Speaker diarization for multiple-distant-microphone meetings using several sources of information," IEEE Transactions on Computers, vol. 56, no. 9, pp. 1212–1224, 2007.

[5] T. Hieu Nguyen, E. S. Cheng, and H. Li, "T-test distance and clustering criterion for speaker diarization," In Proc. Interspeech pp.36-39, 2008.

[6] S. Meignier, D. Moraru, C. Fredouille, J.-F. Bonastre, and L. Besacier, "Step-by-step and integrated approaches in broadcast news speaker diarization," Computer Speech and Language, vol. 20, no. 2-3, pp. 303–330, 2006.

[7] S. Meignier, T. Merlin, "LIUM SpkDiarization: An Open Source Toolkit For Diarization," in Proc. CMU SPUD Workshop, 2010.

[8] H. Cucu, "Towards a speaker-independent, large-vocabulary continuous speech recognition system for Romanian", PhD Thesis, University "Politehnica" of Bucharest, 2011.

[9] http://cmusphinx.sourceforge.net/, last accessed 26.12.2013.