SYSTEMS FOR ROBUST SPEECH ACTIVITY DETECTION AND THEIR RESULTS WITH THE RT05 AND RT06 EVALUATION TESTS

Dušan Macho, Andrey Temko, and Climent Nadeu

TALP Research Center, UPC, Barcelona, Spain
{dusan,temko,climent}@talp.upc.edu

ABSTRACT

Robust Speech Activity Detection (SAD) systems are required in smart-room environments due to the presence of noises and reverberation. In this work, a previous SAD system, based on LDA-extracted features and a decision tree classifier, has been modified in terms of both feature extraction and classification to significantly improve its performance. New features based on the low- and high-frequency energy dynamics, and classifiers based on SVM and GMM have been investigated. In particular, a specific training process has been developed for the SVM case to cope with the problems of that classifier in our application. The resulting SAD systems have been trained with a subset of the SPEECON database. Tested in realistic conditions with the meeting databases from the NIST RT05 and RT06 evaluations, they have shown large improvements in speech detection performance.

1. INTRODUCTION

Detecting the presence of speech is a key objective in speech-related technologies. In fact, Speech Activity Detection (SAD) usually allows an increase of recognition rate in automatic speech or speaker recognition, and it is also required in both speech/speaker recognition and speech coding to save computational resources (and batteries) in the devices where the processing of non-speech events is not needed. Also, as many speech enhancement techniques require a proper estimate of noise characteristics, the reliable detection of non-speech portions of signal is needed. On the other hand, SAD may boost the performance measures of other technologies belonging to audio scene analysis, like speaker localization or acoustic event detection. Last but not least in perceptive interface technologies, the determination of speech activity in a room environment may be used to infer the type of activity that takes place in the room, or at a specific position of the room, given the coordinates of the microphones whose signals show the presence of speech utterances.

Our work, inserted in the CHIL (Computers in the Human Interaction Loop) project framework, assumes a meeting room environment, where audio acquisition is done in an unobtrusive way by a network of far-field microphones. In such a challenging environment, a high robustness of the SAD algorithm against environmental noises and reverberation is extremely important. On the other hand, the working scenarios require online implementations that can operate in real time and only a given maximum latency is accepted. Consequently, segmentation algorithms that use the entire recorded file must be avoided.

In a previous work at our laboratory, we proposed a SAD algorithm [1] that assumed this kind of environment and working conditions. It was compared with other previously reported techniques using a subset of the SPEECON database [2]. The speech detection system was based on speech features that had already shown good robustness properties in automatic speech recognition: the Frequency-Filtered (FF) log spectral energies. The fact that these features are also used for speech recognition avoids the need to re-compute them for SAD when both tasks are being performed at the same time. The FF parameters were further processed by Linear Discriminant Analysis (LDA) to select only one feature per frame, and a Decision Tree (DT) classifier used a time sequence of these features to make the Speech/Non-Speech decision.

In this paper, further work is presented along that line. The already existing algorithm has been tested in more real conditions involving interactions of several persons in meetings and it has been modified to significantly improve its performance. We trained our SAD system with the previous subset of SPEECON and, without any additional tuning, we have used it to carry out tests with the meeting databases from the NIST Rich Transcription 2005 (RT05) evaluation. Both the usual NIST metrics and the ones used in CHIL for SAD have been used to compare performances. In order to improve the SAD results, we have considered two additional features which are measures of energy dynamics at low and high frequencies, respectively. Besides that, two alternative classifiers have been tested, which are based on Support Vector Machines (SVM) [7] and Gaussian Mixture Models (GMM) [9], respectively. The usual training algorithm of the SVM-based classifier has been improved in order to cope with two problems of that classifier in our application: the very large amount of training data and the particular characteristics of the NIST metric. Also, a variant of the GMM-based SAD system was used in the NIST RT06 evaluation campaign, and its results are reported in this paper.

The databases for training and testing are presented in Section 2. Section 3 describes the features and Section 4 is dedicated to the classifier training procedures. Experiments

and results are presented in Section 5, along with the improvements in the training of the SVM-based classifier.

2. DATABASES

For the classifier training, we used a portion of the office environment recordings from the *Spanish* language SPEECON database [2]. In total, 90 minutes of signal recorded by a far-field omni-directional microphone placed 2-3 meters in front of the speaker was used. The training material was well balanced in terms of the two classes of interest; it contained 49% of Speech and 51% of Non-Speech. The database sampling frequency was 16 kHz and the sample representation is 16 bits. Across all recordings, the audio signal uses about 50% of the available 16-bit dynamic range.

For the classifier test/development, we used the single distant microphone evaluation data from the NIST Rich Transcription 2005 (RT05) "conference room" meeting task [3]. It contains 10 extracts from 10 *English* language meetings recorded at 5 different sites. Each extract is about 12 minutes long. The proportion of Speech / Non-Speech is highly unbalanced, approximately 90% of all signal is Speech. The sampling frequency and sampling representation are the same as in the training data, 16 kHz and 16 bits, respectively. Some extracts, however, use only a small portion of the available dynamic range (less than 20%).

The Rich Transcription 2006 (RT06) test data set consists of two kinds of data, "confmtg" and "lectmtg". The confmtg data set is similar to the previously described RT05 data. The lectmtg data were collected from lectures and interactive seminars across the smart-rooms of different CHIL project partners.

The training and development/testing data are similar in a way that they are recorded in a closed environment using a far-field microphone, thus the recordings have a relatively low SNR due to the reverberation and the environmental noise. However, there are some differences that should be mentioned: different language (Spanish vs. English), different setup of the acquisition hardware, different Speech and Non-Speech proportion. Also, it is worth to mention that the main task, and thus the main attention, of the speaker in the training database was the recording itself, while in the test meeting/lecture database, the recording was secondary. As a consequence, the test/lecture database is more spontaneous, speakers speak not necessarily heading the microphone, and the data contain overlapped speech.

3. FEATURES

We investigate two kinds of features. The first feature set, based on Linear Discriminant Analysis (LDA) [9] of parameters that model spectra, extracts the information about the spectral shape of the acoustic signal from a short interval (approx. 70 ms). The second feature set focuses more on the dynamics of the signal along the time observing low- and high-frequency spectral components along a bit longer time interval (approx. 130 ms).

3.1. LDA Measure

The LDA measure, *ldam*, is based on Frequency Filtering (FF) features – a speech representation originally designed for ASR that showed higher robustness in noisy ASR tests than the usual mel-frequency cepstrum (MFCC) features (see e.g. [4]). The robustness issue is very important due to the low SNR of the recordings in our task.

The FF feature extraction scheme used in this work consists in calculating a log filter-bank energy vector of 16 bands for each signal frame (with frame length/shift = 30/10ms) and then applying a FIR filter with impulse response $h(k)=\{1, 0, -1\}$ on this vector along the frequency axis. The obtained static FF feature vector is accompanied with a short-time dynamic representation in form of delta (50ms) and delta-delta (70ms) features. In addition, the delta of the frame energy is also appended. The size of the FF representation (16+16+16+1=49) is reduced to a single scalar measure by applying LDA, a data-driven linear transformation designed to extract the principal components of the input data using a discriminative criterion. That single LDA measure, *ldam*, is computed by multiplying the FF feature vector and the LDA eigenvector corresponding to the largest LDA eigenvalue as calculated from the training set. More details on the LDA FF features can be found in [1], where it is also shown that the FF+LDA measure is more discriminative than the MFCC+LDA measure.

3.2. Low-Frequency and High-Frequency Energy Dynamics Feature

In addition to the LDA measures, we experimented with two sub-band energy based features, low-frequency and high-frequency energy dynamics feature (*lfed* and *hfed*, respectively). *lfed* is calculated as follows:

$$E_l(t) = \log\left(\sum_k S(k,t)\right) \text{ where } 13 \le k \le 38 \quad (1)$$

$$dE_l(t) = \frac{1}{60}\sum_{i=-4}^{4} i \cdot E_l(t+i) \quad (2)$$

$$lfed(t) = \frac{1}{5}\sum_{i=-2}^{2} abs\left(dE_l(t+i)\right) \quad (3)$$

where $S(k,t)$ is the k-th bin of the FFT-512 power spectrum at the frame index t. *lfed* involves approximately a frequency range from 400 Hz to 1200 Hz comprising most of the interval of high energy concentration of the voiced speech sounds (sampling frequency of 16 kHz is assumed). *hfed* is

calculated in the same way but $144 \le k \le 208$, which correspond to the interval from 4500 Hz to 6500 Hz and this feature focuses on fricative sounds. The frequency intervals of both features are based on general knowledge and were not tuned to the application. A similar feature as *lfed* and *hfed* was proposed in [5] but that feature was calculated over the entire frequency range and it included spectral autocorrelation to emphasize the speech harmonic structure.

Notice that in the final signal representation, the contextual information is involved in several ways. First, before applying the LDA transform, the current delta and delta-delta feature involve an interval of 50 and 70 ms, respectively, in their calculation. Next, for the representation of the current frame, eight LDA measures are selected from a time window spanning the interval of 310 ms around the current frame. Finally, lfed and hfed involve a smoothed derivative calculation that in total uses an interval of 130 ms.

We use the SAD system on-line in our smart-room, so we avoid using techniques that would cause an algorithmic delay larger than a given acceptable value (set to 160 ms in our case). In addition, the designed SAD feature extraction saves computational resources since most of the calculation dedicated to the feature extraction is performed anyway for the ASR system due to the fact that SAD features are based on ASR features.

4. CLASSIFIER TRAINING

In this section, we explain the training procedures used for the three classifiers used in this work. We use the Decision Tree (DT) as our baseline classifier and we contrast its performance with another discriminative classifier based on Support Vector Machine (SVM) [7] approach and a generative classifier based on Gaussian Mixture Model (GMM) [9].

4.1. Features

For each frame at a time index *t*, one LDA measure *ldam*(*t*) is available. To include information from a time span larger than just 70ms into the representation, the eight most important LDA measures are selected from the interval $t-15 \le t \le t+15$. As a criterion for this selection, we used the entropy based information gain criterion used in the DT training algorithm (see [1] for more details on this selection process). LDA measures were concatenated to form the final representation vector.

ldam(t-15) ldam(t-10) ldam(t-6) ldam(t-3) ldam(t) ldam(t+3) ldam(t+6) ldam(t+10)

Using these features, we defined the following four different feature sets (in parenthesis is the feature vector size):

A: Eight ldam features selected using the C4.5 DT training (8 features)
B: A + lfed (9 features)
C: A + hfed (9 features)
D: B + hfed (10 features)

The A feature set is considered a baseline in our tests (a six-feature version of this set was already used in [1]). The feature sets B, C, and D allow us to observe the contribution of the low and high frequency dynamics features when added individually to the feature set A, as well as when both of them are added to A.

The 90 minute SPEECON training data, processed on frame-by-frame basis using a frame shift of 10 ms, results in over 500 thousand training examples with their corresponding Speech / Non-Speech labels. The Speech / Non-Speech labeling was performed by applying a forced Viterbi alignment on the training files using our speech recognition system.

4.2. Decision Tree (DT) Classifier

For DT training we used the C4.5 algorithm [6] which is an improvement of Quinlan's original ID3 DT training algorithm. During training, for each node of the decision tree, the best feature element from the feature vector is selected and the best threshold is set for this element. Using the SPEECON training example set and setting the pruning confidence level to 25% resulted in decision trees with the following number of nodes, depending on the feature set: A 1031, B 1475, C 1571, and D 2117.

4.3. Support Vector Machine (SVM) Classifier

A set of 500 thousand of examples is an enormous number of feature vectors to be used for the usual SVM training approach and hardly makes such training process feasible in practice. Alternative methods should be used; we tested the so-called cascade learning [8], however our implementation did not achieve a satisfactory performance. A better result was obtained by imposing a hard data reduction by randomly selecting 20 thousand examples where the two classes of interest are equally represented. The training data were firstly normalized anisotropicly to be in the range from –1 to 1, and the obtained normalizing template was then applied also to the testing data set. We used the Gaussian kernel with gamma parameter equal 5.0 and the C parameter (controlling the training error) equal 10.0. To train the system we use the publicly available SVMlight software package [10]. That preliminar SVM system was posteriorly modified as explained in the subsection 5.3.

4.4. Gaussian Mixture Model (GMM) Classifier

We used the well known Expectation-Maximization (EM) [11] algorithm for Gaussian mixture model training with the K-means algorithm for the model parameter initialization. The number of mixtures was set to 32 for both Speech and Non-Speech classes and diagonal covariance matrices were used. 20 iterations of the EM algorithm were performed.

5. EXPERIMENTS

5.1. Metrics

We present results using several metrics; as a primary metric we use the one defined for the SAD task in the NIST Rich Transcription evaluation. It is defined as the ratio of the duration of incorrect decisions to the duration of all speech segments in reference. We denote this metric as NIST in our results.

Notice that the NIST metric depends strongly on the prior distribution of Speech and Non-Speech in the test database. For example, a system that achieves a 5% error rate at Speech portions and a 5% error rate at Non-Speech portions, would result in very different NIST error rates for test databases with different proportion of Speech and Non-Speech segments; in the case of 90-to-10% ratio of Speech-to-Non-Speech the NIST error rate is 5.6%, while in the case of 50-to-50% ratio it is 10%. Due to this fact we report three metrics from CHIL: Mismatch Rate (MR), Speech Detection Error Rate (SDER), and Non-Speech Detection Error Rate (NDER) defined as:

- MR = Duration of Incorrect Decisions / Duration of All Utterances
- SDER = Duration of Incorrect Decisions at Speech Segments / Duration of Speech Segments
- NDER = Duration of Incorrect Decisions at Non-Speech Segments / Duration of Non-Speech Segments

5.2. Results on RT05 testing data

For the RT05 test database, a post-processing was applied to each SAD output consisting of marking the non-speech intervals shorter than 0.3 seconds as speech. This non-speech gap smoothing was used to mimic the same post-processing that was applied to the original human labels used as the reference.

Notice also that the test database contains much more Speech intervals than Non-Speech intervals (approx. 90% Speech). The NIST metric is inversely proportional to the amount of Speech in the reference labels. Thus, assuming the same amount of incorrect decisions in both a testing data with 50% Speech content and a testing data with 90% Speech content, a lower NIST error will be reported for the later testing data. This is the reason why some error rates reported for the unbalanced test database (RT05) may be lower that those reported for the more balanced train database (SPEECON).

Table 1 shows the results we obtained on the test database. Neither features nor classifiers were tuned to these test data. Most of the observations about the lfed and hfed features from the previous experiments with the training data hold also in this case, which is quite encouraging. An exception is the feature set D in the GMM classifier, where adding hfed does not improve the performance of the feature set B for the same classifier. In general, significant improvements can be seen when adding the lfed and hfed features to the LDA vector; for example in the case of SVM, the usage of both features reduces the original error by 52%. Among the classifiers, GMM achieves substantially lower error rates than the two discriminative classifiers (note that GMM was the worst performing on the training data). It seems in our case that the GMM classifier generalizes better the knowledge from the training data than the two other classifiers. The performance of DT and SVM is very similar, especially in the D feature set case. The best overall performance, NIST error of 8.47%, was obtained using the GMM classifier with the feature set B. It represents a 59% error rate reduction with respect to the performance of the baseline system consisting of the DT classifier and the feature set A (20.69%). Notice the NDER scores are high in comparison to the SDER scores in these tests. This is caused mostly by the combined effect of the Non-Speech gap post-processing and the low amount of the Non-Speech testing material which was mentioned above.

Table 1. Error rates obtained for the RT05 test data

	NIST MR / SDER / NDER			
Feat set	A	B	C	D
DT	20.69 18.77 / 18.21 / 24.32	12.37 11.20 / 9.24 / 30.51	14.76 13.37 / 11.65 / 30.27	11.54 10.43 / 8.10 / 33.42
SVM	23.88 21.71 / 21.46 / 24.22	14.70 13.36 / 11.87 / 28.19	15.69 14.26 / 12.51 / 31.76	11.45 10.41 / 7.99 / 34.56
GMM	12.25 11.13 / 8.86 / 33.81	8.47 7.69 / 4.61 / 38.42	10.02 9.11 / 5.24 / 47.70	8.66 7.88 / 3.75 / 49.00

5.3. Improvement of the SVM performance on the RT05 testing data

As it has been mentioned in subsection 4.3, a hard data reduction method has been applied that randomly selects 20 thousand examples from the SPEECON database where the two classes of interest are equally represented. The experiments in this subsection aim to improve the

performance of the SVM classifier doing two modifications in the training process.

The first one aims at an efficient sample selection. Instead of using a random selection of a reasonable number of samples, a two-step approach is chosen. Firstly, the whole training database is decomposed into chunks of 1000 samples. Then, on each chunk, a Proximal Support Vector Machine (PSVM) [12] has been trained. Unlike conventional SVM, PSVM solves a single square system of linear equations and thus it is very quick to train. In the nonlinear case (we use a Gaussian kernel) of PSVM the concept of support vector disappears as the separating hyperplane depends on all chunk data. In that way, all training data must be preserved for testing. While training, PSVMs 5-fold cross validation (CV) was applied to obtain optimal C and gamma parameters. After training, a threshold was applied on the CV accuracies of all chunks to select a given number of them that show the highest CV accuracy (we select the same number of data as in subsection 4.3, i.e. 20 chunks = 20 thousand samples). In the second step, a conventional SVM with the setting described in subsection 4.3 is trained on the chosen data.

The second modification makes use of the knowledge of the specific NIST metrics during the training phase. As it has been mentioned in subsection 5.1, NIST metrics depends on the prior distribution of Speech and Non-Speech in the test database. For this reason, if we want to improve the NIST scores for the RT05 evaluation, we should penalize the Speech class more than the Non-Speech class. That is possible for a discriminative classifier as SVM, by introducing different costs for the two classes (but not for a GMM-based classifier).

Table 2 shows results obtained on the RT05 evaluation with the modified SVM system and feature set D, along with the ones obtained with the best SVM and GMM systems from Table 1.

Table 2. Error rates obtained for the RT05 evaluation with the modified SVM system

	NIST MR / SDER / NDER
GMM	8.47 7.69 / 4.61 / 38.42
SVM	11.45 10.41 / 7.99 / 34.56
SVM modified	8.03 7.30 / 2.51 / 55.07

From Table 2 we observe that, as it can be expected after the second modification, the NDER score has increased but the SDER score, which has the major influence on the NIST measure, has strongly decreased. In consequence, after both modifications, the NIST error for the SVM-based system decreases from 11.45% to 8.03%, slightly outperforming the best GMM system.

5.4. Results on RT06 testing data

In this subsection we present the results achieved in the RT06 evaluation campaign. For that evaluation we chose a GMM classifier and, as features, we used the feature set D augmented by a cross-frequency energy dynamic feature, *xfed*, which is obtained as a combination of *lfed* and *hfed* and it is calculated as follows:

$$xfed(t) = \frac{\sqrt{hfed(t-9)\cdot lfed(t+9)}}{2} + \frac{\sqrt{hfed(t+9)\cdot lfed(t-9)}}{2} \quad (4)$$

This feature reaches high values when both the *hfed* before and *lfed* after (or *hfed* after and *lfed* before) the current frame have high values. It attempts to follow the energy flow between low and high frequencies typical for speech. The 9 frame distance was set empirically (it would correspond to 1/0.18 = 5.6 Hz energy flow rate) and it is limited by the maximum allowed algorithmic delay of the SAD system.

For the confmtg task, both SPEECON and RT05 databases were used to train the SAD system. For the lectmtg task, also a small amount of CHIL data was added into the training of the final system.

The RT06 evaluation campaign has several evaluation subtasks, depending on the used set of microphones. We participated in two subtasks, single distant microphone denoted as *sdm* and multiple distant microphone, *mdm*. The *sdm* subtask involved the centrally located omni-directional table microphone, or the best microphone selected after listening to the recordings. The *mdm* subtask involved at least 3 omni-directional table microphones, including the one selected for the *sdm* subtask.

In the tests with the RT06 data we applied a slightly different post-processing on top of the classifier output than we did in the case of the RT05 data tests. First, we performed a majority voting, where the central frame of an 11 frame interval was marked as Speech if 6 or more frames of this interval were classified as Speech; otherwise it was marked as Non-Speech. Second, to each Speech segment, 0.2 s of Speech was added at the beginning and the end of segment.

In the case of mdm, we applied a separate SAD to each individual channel, without post-processing. Outputs of all SAD systems were merged by a majority voting performed for each frame favoring the Speech label in the case of a tie. Then the same post-processing as in the sdm case was applied on the output of merging.

Table 3. Error rates obtained for the RT06 evaluation tasks

Feat set D + *xfed*	NIST MR / SDER / NDER	
	confmtg	lectmtg
sdm	5.45 5.1 / 3.1 / 41.4	7.10 6.2 / 0.4 / 48.1
mdm	5.63 5.3 / 3.5 / 38.7	5.30 4.6 / 0.7 / 33.3

Table 3 shows error rates obtained by our SAD system in the RT06 evaluation tasks. Very low NIST error rates were obtained and our SAD system ranked among the best systems; however, the Non-Speech detection error rates are high. To reduce this error rate will be the objective of our future work. We could benefit from the multiple microphones in the case of lectmtg task, however there is no significant change in the case of confmtg task.

6. CONCLUSION

The presented work is oriented towards a robust Speech Activity Detection (SAD) in smart-room environments. The baseline SAD system used features obtained by applying Linear Discriminative Analysis (LDA) on a parameter set modeling the shape of the signal spectrum; Decision Tree was used to perform the Speech/Non-Speech classification on a frame-by-frame basis. Both the LDA transform and the Decision Tree classifier were trained by a portion of the Spanish SPEECON database. The SAD system was evaluated on an interactive meeting database from the NIST RT05 evaluation campaign.

In this work we improved significantly the performance of the baseline speech detector. We tested additional features that measure the signal energy dynamics at low and high frequencies. Also, two other classifiers were evaluated for the SAD task: a discriminative Support Vector Machine classifier and a generative Gaussian Mixture Model (GMM) classifier. We observed that appending the high and low frequency energy dynamics features to the LDA features improved the performance for both training and testing data across all the classifiers with a higher benefit from the low-frequency feature.

In a first stage, the highest NIST error rate reduction was achieved by using the GMM classifier and the LDA features with the low-frequency energy dynamics; the error of the baseline SAD was reduced from 20.69% to 8.47% in this case. Then, two modifications of the usual training algorithm of the SVM-based classifier were developed in order to cope with two problems of that classifier in our application: the very large amount of training data and the particular characteristics of the NIST metric. With those two modifications, the SVM system further reduced the error to 8.03%. Finally, very competitive results were obtained in the RT06 SAD evaluation task, and they were also reported in this paper.

7. ACKNOWLEDGEMENTS

The authors wish to thank Jaume Padrell for encouraging discussions on the topic. This work has been partially sponsored by the EC-funded project CHIL (IST-2002-506909) and the Spanish Government-funded project ACESCA (TIN2005-08852).

8. REFERENCES

[1] Padrell J., Macho D., Nadeu C., "Robust Speech Activity Detection Using LDA Applied to FF Parameters", Proc. ICASSP'05, Philadelphia, PA, USA, March 2005.

[2] Iskra D. J. et al., "SPEECON - Speech Databases for Consumer Devices: Database Specification and Validation", Proc. LREC, 2002.

[3] Fiscus J.G., Radde N., Garofolo J.S., Le A., Ajot J., Laprun C., "The Rich Transcription 2005 Spring Meeting Recognition Evaluation", Lecture Notes in Computer Science (LNCS), vol. 3869, pp.369-389, Springer, February 2006.

[4] Nadeu, C., Macho, D., and Hernando, J., "Frequency and Time Filtering of Filter-Bank Energies for Robust HMM Speech Recognition", Speech Communication, Vol. 34, pp. 93-114, 2001.

[5] Ouzounov A., "Robust Feature for Speech Detection", Cybernetics and Information Technologies, vol.4, No.2, pp.3-14, 2004.

[6] Quinlan, J. R., *C4.5: Programs for Machine Learning*, Morgan Kaufmann, 1992.

[7] B. Schölkopf, A. Smola, *Learning with Kernels*, MIT Press, Cambridge, MA, 2002.

[8] Graf H.P., Cosatto E., Bottou L., Durdanovic I., Vapnik V., "Parallel Support Vector Machines: The Cascade SVM", Proc. Eighteenth Annual Conference on Neural Information Processing Systems, 2004.

[9] Duda R., Hart P., Stork D., *Pattern Classification*, 2nd Edition, Wiley-Interscience, 2000.

[10] SVMlight: http://svmlight.joachims.org/

[11] Rabiner L., Juang B.H., *Fundamentals of Speech Recognition*, Prentice Hall, 1993.

[12] Fung G., Mangasarian O., "Proximal Support Vector Machine Classifiers", Proc. Int. Conference on Knowledge Discovery and Data Mining, pp. 77-86, 2001.