Odyssey 2016
21-24 Jun 2016, Bilbao, Spain

I-Vectors for Speech Activity Detection

Elie Khoury, Matt Garland

Pindrop, Atlanta, USA
{ekhoury,mgarland}@pindropsecurity.com

Abstract

I-Vectors are low dimensional front-end features known to effectively preserve the total variability of the signal. Motivated by their successful use for several classification problems such as speaker, language and face recognition, this paper introduces i-vectors for the task of speech activity detection (SAD). In contrast to most state-of-the-art SAD methods that operate at the frame or segment level, this paper proposes a cluster-based SAD, for which two algorithms were investigated: the first is based on generalized likelihood ratio (GLR) and Bayesian information criterion (BIC) for segmentation and clustering, whereas the second uses K-means and GMM clustering. Furthermore, we explore the use of i-vectors based on different low-level features including MFCC, PLP and RASTA-PLP, as well as fusion of such systems at the decision level. We show the feasibility and the effectiveness of the proposed system in comparison with a frame-based GMM baseline using the challenging RATS dataset in the context of the 2015 NIST OpenSAD evaluation.

1. Introduction

Speech Activity detection (SAD) aims to distingush between speech and non-speech (e.g. silence, noise or music) regions within audio signals. SAD is an important and necessary preprocessing step in a number of applications such as speaker recognition and diarization, language recognition, and speech recognition. It is also used to assist humans in analyzing recorded speech for applications such as forensics, enhance speech signals, and improve compression of audio streams before transmission.

Existing SAD techniques fall into two categories: supervised and unsupervised [1]. Among the supervised techniques, GMM [2] is perhaps the most widely used. Motivated by the success of i-vectors [3] over GMMs on several classification tasks such as speaker and language recognition, this work presents, to the best of our knowledge, the first attempt to apply i-vectors for SAD.

Most existing SAD approaches operate at the frame level. This makes them subject to high smoothing error and highly dependent on window-size tuning. In contrast, we propose a cluster-level SAD. Two algorithms are investigated: the first is based on generalized likelihood ratio and Bayesian information criterion (GLR/BIC) for segmentation and clustering, whereas the second is based on K-means and GMM clustering. Clustering is suitable for i-vectors since only a single i-vector is extracted per cluster, and this approach avoids the computational cost of extracting i-vectors on overlapped windows, in contrast to existing approaches that use contextual features [4, 5]. Two different classification techniques are explored for discriminating between the speech and non-speech i-vectors: probabilistic linear discriminant analysis (PLDA) [6] and support vector machine (SVM) [7]. Fig. 1 illustrates the scheme of the proposed SAD systems.

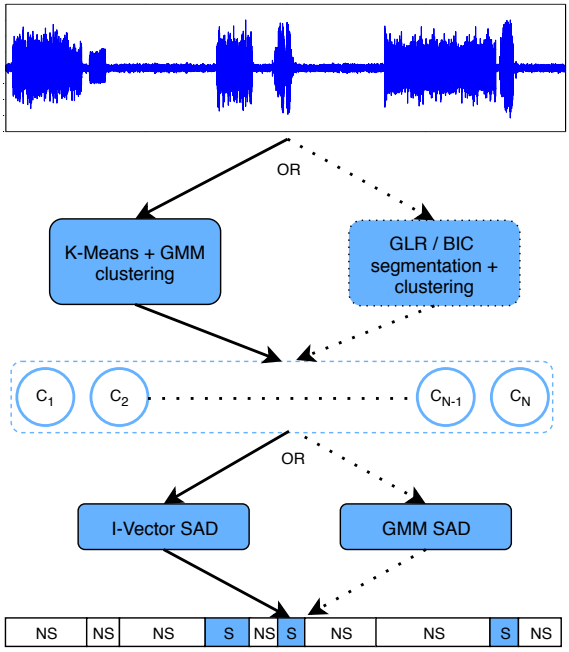

Figure 1: Proposed speech activity detection system. It includes a first step of feature clustering using either K-means and GMM (left side) or GLR/BIC (right side) and a second step of SAD based on either i-vectors (left side) or GMM (right side).

Different audio features were considered in our study, namely MFCC, PLP and RASTA-PLP. In addition, we applied a score-level fusion based on logistic regression to combine decision outputs from different SAD systems. Experiments were carried out on the RATS dataset [8] in the context of the 2015 NIST OpenSAD challenge[1].

The remainder of this paper is organized as follows: Section 2 reviews the state-of-the-art in speech activity detection. Section 3 presents the different clustering techniques used in this work. Section 4 describes both cluster-based GMM and i-vector classifiers used for SAD. Section 5 details the experimental setup and results. Section 6 concludes the paper.

[1]NIST disclaimer: "NIST serves to coordinate the NIST OpenSAD evaluations in order to support speech activity detection research and to help advance the state-of-the-art in speech activity detection technologies. NIST OpenSAD evaluations are not viewed as a competition: as such, results reported by NIST are not to be construed, or represented, as endorsements of any participant's system, or as official findings on the part of NIST or the U.S. Government". Web page: http://www.nist.gov/itl/iad/mig/opensad_15.cfm

doi: 10.21437/Odyssey.2016-48

2. Related Work

A wide spectrum of approaches exist in the literature to address speech activity detection. They range from very simple systems such as energy-based classifiers to extremely complex ones such as deep neural networks (DNN). Although the SAD task is very old, recent studies on real-life data have shown that state-of-the-art SAD techniques lack generalization power. This explains the increased research interest in the last few years, especially within the DARPA RATS program[2].

Existing SAD approaches can be categorized into unsupervised and supervised techniques. **Unsupervised SAD techniques** include standard real-time SADs such as the one used by G.729 [9] in telecommunication products (e.g. voice over IP). To meet the real-time requirements, these techniques combine a set of low-complexity, short-term features such as spectral frequencies, full-band energy, low-band energy, and zero-crossing rate extracted at the frame level (10 ms). The classification between speech and non-speech is made using either hard or adaptive thresholding rules.

More robust unsupervised techniques assume access to long-duration buffers (e.g. multiple seconds) or even the full audio recording. This helps to improve feature normalization and gives more reliable estimates of statistics. Examples of such techniques are energy-based bi-Gaussians, vector quantization [10], 4Hz modulation energy [11], *a posteriori* signal-to-noise ratio (SNR) weighted energy distance [12], and unsupervised sequential GMM applied on 8-Mel sub-bands in the spectral domain [13].

Although unsupervised approaches do not require any training data, they often suffer from relatively low detection accuracy compared to supervised approaches. One main drawback is that they are highly dependent on the balance between speech and non-speech regions (e.g. energy-based bi-Gaussian technique).

Supervised SAD techniques include Gaussian mixture models (GMM) [1, 14, 15, 16], hidden Markov model (HMM) Viterbi segmentation [4], deep neural network (DNN) [5], recurrent neural network (RNN) [17], and long short-term memory (LSTM) RNN [18].

Different acoustic features may be used in supervised approaches, varying from standard features computed on short-term windows (e.g. 20 ms) such as MFCC, PLP, RASTA-PLP, or power-normalized cepstrum coefficients (PNCC) [16] to more sophisticated long-term features that involve contextual information such as frequency domain linear prediction (FDLP), voicing features, and Log-mel features [4, 19].

Supervised methods use training data to learn their models and architectures. They typically obtain very high accuracy on seen conditions in the training set but fail in generalizing to unseen conditions. Moreover, they are more complex to tune and time consuming, especially during the training phase.

One common drawback of most existing supervised and unsupervised SAD approaches is that their decisions operate at the frame level (even in the case of contextual features), which cannot be reliable by itself, especially at boundaries between speech and non-speech regions [17]. Smoothing techniques are often used to alleviate this issue.

To reduce the effect of such problems, this work proposes a SAD technique where the decision is made at the cluster level instead of the frame level and is thus more robust to the local behavior of the features.

[2] http://www.darpa.mil/program/robust-atuomatic-transcription-of-speech

3. Data Structuring

3.1. GLR/BIC Segmentation and Clustering

The goal of this task is to split the audio recording into a set of segments S_i where each segment ideally contains only one audio source, then merge the most similar segments in a hierarchical bottom-up manner. This technique is inspired by the state-of-the-art work on speaker diarization [20, 21, 22, 23].

Let $X = x_1, \ldots, x_{N_x}$ be a sliding window of feature vectors of dimension d and M its parametrical model. We assume M to be multivariate Gaussian. The feature vectors are either MFCC, PLP or RASTA-PLP extracted on 20 ms windows with a shift of 10 ms. In practice, the size of the sliding window X is empirically set to 1 second (i.e. $N_X = 100$).

The generalized likelihood ratio (GLR) [20] is used to select one of two hypotheses:

- H_0 assumes that X belongs to only one audio source. Thus, it is best modeled by a single multivariate Gaussian distribution:

$$(x_1, \ldots, x_{N_x}) \backsim N(\mu, \sigma) \tag{1}$$

- H_c assumes that X is shared between two different audio sources separated by a point of change c: the first source is in $X_{1,c} = x_1, \ldots, x_c$ whereas the second is in $X_{2,c} = x_{c+1}, \ldots, x_{N_x}$. Thus, the sequence is best modeled by two different multivariate Gaussian distributions:

$$(x_1, \ldots, x_c) \backsim N(\mu_{1,c}, \sigma_{1,c}) \tag{2}$$

and

$$(x_{c+1}, \ldots, x_N) \backsim N(\mu_{2,c}, \sigma_{2,c}) \tag{3}$$

Therefore, GLR is expressed by:

$$GLR(c) = \frac{P(H_0)}{P(H_c)} = \frac{L(X, M)}{L(X_{1,c}, M_{1,c})L(X_{2,c}, M_{2,c})} \tag{4}$$

where $L(X, M)$ is the likelihood function. Considering the log scale , $R(c) = ln(GLR(c))$, Eq. 4 becomes:

$$R(c) = \frac{N_X}{2} \log|\Sigma_X| - \frac{N_{X_{1,c}}}{2} \log|\Sigma_{X_{1,c}}| - \frac{N_{X_{2,c}}}{2} \log|\Sigma_{X_{2,c}}| \tag{5}$$

where Σ_X, $\Sigma_{X_{1,c}}$ and $\Sigma_{X_{2,c}}$ are the covariance matrices and N_X, $N_{X_{1,c}}$ and $N_{X_{1,2}}$ the number of vectors of X, $X_{1,c}$ and $X_{2,c}$, respectively. A Savitzky-Golay filter [24] is applied to smooth the $R(c)$ curve Example output of such filtering is presented in Fig. 2(b).

By maximizing the likelihood, the estimated point of change $\widehat{c}_{\text{glr}}$ is:

$$\widehat{c}_{\text{glr}} = arg\max_c R(c) \tag{6}$$

The above GLR algorithm detects a first set of candidates for segment boundaries, which are then used in a stronger detection phase based on Bayesian information criterion (BIC) [21]. The goal of BIC is to filter out the points that are falsely detected and to adjust the remaining points. The new segments boundaries are estimated as follows:

$$\widehat{c}_{\text{bic}} = arg\max_c \Delta BIC(c) \tag{7}$$

where

$$\Delta BIC(c) = R(c) - \lambda P \tag{8}$$

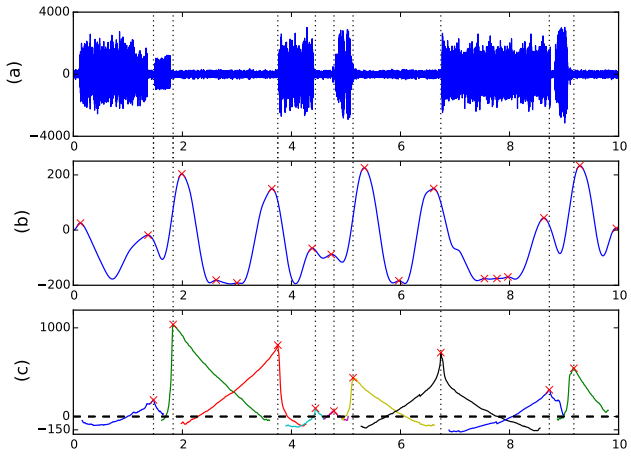

Figure 2: GLR and BIC automatic responses. Sub-figure (a) illustrates 10 seconds of an audio signal. Sub-figure (b) shows the curve produced by GLR. It also shows the first set of segment boundaries (b) that correspond to local minima on the curve. Sub-figure (c) shows the refinement power of BIC, where boundaries are accurately detected. The colored curves show the variation of Eq. 8 on a variable-size shifted window.

and preserved if $\Delta BIC(\widehat{c}_{\text{bic}}) \geq 0$. As shown in Eq. 8, the BIC criterion derives from GLR with an additional penalty term λP that depends on the size of the search window [25].

Fig. 2(a) plots a 10-second audio signal. The actual responses of smoothed GLR and BIC are shown in Fig 2(b) and Fig 2(c), respectively. The colored curves in Fig 2(c) correspond to Eq.8 applied on a single window each. The local maxima are the estimated boundaries of the segments and accurately match the ground truth.

Finally, the resulting segments are grouped by hierarchical agglomerative clustering (HAC) and the same BIC distance measure [26] used in Eq. 8. We avoid unbalanced clusters by introducing a constraint on the size of the clusters, and the stopping criterion chosen is when all clusters have duration higher than D_{min}. D_{min} is empirically set to 5 seconds.

3.2. K-means + GMM Clustering

The K-means and GMM clustering is the typical clustering used for universal background model (UBM) training in the GMM-based speaker and language recognition systems. It is accomplished using the *Expectation - Maximization* (EM) [27] algorithm to maximize the likelihood over all the features of the audio recording. This partitional clustering is faster than the hierarchical clustering and does not require a stopping criterion; however, it requires the number of clusters (K) to be set in advance. In our experiments, we choose K to be dependent on the duration of the full recording $D_{\text{recording}}$:

$$K = \left\lceil \frac{D_{\text{recording}}}{D_{\text{avg}}} \right\rceil + 1 \tag{9}$$

where D_{avg} is the average duration of the clusters and $\lceil\rceil$ denotes the ceil. D_{avg} is empirically set to 5 seconds. It is worth noting that the minimum number of clusters in Eq. 9 is two. This makes SAD possible for utterances shorter than D_{avg}.

4. Classifiers for Speech Activity Detection

Both clustering algorithms result in a set of clusters that are highly pure (see Table 1). Each of these clusters $\mathcal{C}$ is mostly one type i of data:

$$i \in \{\text{Speech}, \text{NonSpeech}\} \tag{10}$$

The following sections present two classification techniques, namely GMM and I-Vectors, and the score-level fusion approach.

4.1. Gaussian Mixture Models

To use GMMs for SAD, we need to learn a GMM $\mathcal{G}_i$ for each type i from a set of enrollment samples. As in [2], the training is done using the EM algorithm to seek a *maximum-likelihood* estimate. Once type-specific models $\mathcal{G}_i$ are trained, the probability that a test cluster $\mathcal{C}_t$ is from the class Speech is given by a *log-likelihood ratio* (LLR) score:

$$h_{\text{gmm}}(\mathcal{C}_t) = \ln p(\mathcal{C}_t|\mathcal{G}_{\text{Speech}}) - \ln p(\mathcal{C}_t|\mathcal{G}_{\text{NonSpeech}}) \tag{11}$$

It is worth noting that the frame-based baseline system used in our experiments can be viewed as a special case of the above formulation with $\mathcal{C}_t$ being one single frame.

4.2. I-Vectors

Total variability modeling aims to extract low-dimensional factors $w_{i,j}$, so-called *i-vectors*, from samples $\mathcal{C}_{i,j}$, using the following expression:

$$\mu = m + T\omega \tag{12}$$

where μ is the supervector of $\mathcal{C}_{i,j}$, m is the supervector of universal background model, T is the low-dimensional total variability subspace, and ω the low-dimensional i-vector, which is assumed to follow a normal distribution $\mathcal{N}(\mathbf{0}, I)$.

The procedure to learn the total variability subspace T relies on EM algorithm that maximizes the likelihood over the training set of labeled speech and non-speech segments.

Once i-vectors are extracted, whitening and length-normalization [28] are applied for channel compensation purposes. Finally, we tried two back-end classifications: PLDA [6] and SVM [7]. For PLDA, the LLR of a test cluster $\mathcal{C}_t$ being from the class Speech is expressed as:

$$h_{\text{plda}}(\mathcal{C}_t) = \frac{p(w_t, w_{\text{Speech}}|\Theta)}{p(w_t|\Theta)p(w_{\text{Speech}}|\Theta)} \tag{13}$$

where w_t is the test i-vector, w_{Speech} the mean of speech i-vectors, and $\Theta = \{F, G, \Sigma_\epsilon\}$ is the PLDA model. F and G are the between-class and within-class covariance matrices and Σ_ϵ is the covariance of the residual noise.

For SVM, we used the Platt scaling [29] to transform SVM scores into probability estimates:

$$h_{\text{svm}}(\mathcal{C}_t) = \frac{1}{1 + \exp(Af(w_t) + B)} \tag{14}$$

where $f(w_t)$ is the uncalibrated score of the test sample obtained from SVM [7], and A and B are learned on the training set using maximum likelihood estimation. In our experiments, we used SVM with a radial basis function (RBF) kernel, which we found to work better than a linear kernel.

4.3. Fusion

MFCC, PLP, and RASTA-PLP features were studied in our experiment to assess the generality of our proposed method. We also applied a score-level fusion over the different features' individual SAD systems to evaluate whether cluster-based SAD provides any incremental benefit over frame-based SAD. Towards this end, we use *logistic regression* approach that has been successfully employed for combining heterogeneous speaker classifiers [30]. Let a test utterance $\mathcal{C}_t$ be processed by N_s SAD systems. Each system produces an output score denoted by $h_s(\mathcal{C}_t)$. The final fused score is expressed by the logistic function:

$$h_{\text{fusion}}(C_t) = g\left(\alpha_0 + \sum_{s=1}^{N} \alpha_s h_s(C_t)\right) \tag{15}$$

where

$$g(x) = \frac{1}{1+\exp(-x)} \tag{16}$$

and $\alpha = [\alpha_0, \alpha_1, \ldots, \alpha_N]$ are the regression coefficients.

5. Experimental Evaluation

5.1. Experimental Setup

DARPA Robust Automatic Transcription of Speech (RATS) program [8] is designed to advance the state-of-the-art speech activity detection in distorted, degraded and noisy communication channels. Different frequency bands (HF, UHF and VHF) and different modulation types (narrow-band FM, wide-band FM, AM, frequency-hopping spread-spectrum and SSB) from the RATS program are considered in this study.

OpenSAD is an evaluation organized by NIST on part of the RATS dataset. More precisely, six channels (B, D, E, F, G and H) from the RATS were used in the training and development sets and two additional channels (A and C) in the evaluation set.

The training set used to train background models contains 5,485 audio recordings consisting of 1071 hours of data. The results reported in this study are solely based on the development set (part 2) that contains 661 audio recordings consisting of 169 hours of data. On this set, the average duration of an audio recording is 15 minutes and 19 seconds, with speech regions comprising 35.12% of the audio.

The evaluation metric used in OpenSAD is the minimum detection cost function, given by:

$$\text{minDCF} = \gamma\text{FAR} + (1-\gamma)\text{FRR} \tag{17}$$

where FAR is the false alarm rate and FRR is the false rejection rate. The weight γ is set to 0.25 to penalize the missed detection of speech more heavily. While the official evaluation metric in OpenSAD allows a 2-second collar around speech regions, we consider a strict protocol with no collar that is more adequate for applications such as speaker and language recognition. The strict protocol also avoids any uncontrolled bias introduced by the collar factor. In addition, we impose a global threshold to make SAD systems as channel-independent as possible.

The hyper-parameters of each structuring technique, namely λ and N_X for the GLR/BIC segmentation, D_{min} for BIC hierarchical clustering, and D_{avg} for K-means and GMM clustering, were tuned to maximize the speech detection accuracy (i.e. to reduce minDCF).

Regarding the SAD classifiers, we found that 32 components for the GMM models (both the Speech and NonSpeech models in the GMM-based system and the UBM model in the I-Vector system) and a rank of 100 for T provide a good trade-off between accuracy and speed.

Table 1: Purity of clusters and accuracy of segmentation, segmentation + HAC clustering, K-means clustering, and K-means + GMM clustering. MFCC features produce the purest clusters under GLR/BIC, while PLP produces the purest clusters under K-means + GMM.

Method	Metric	MFCC	PLP	RASTA-PLP
Segmentation	Purity (%)	**94.5**	94.2	93.6
	minDCF	0.131	0.134	0.142
Segmentation + HAC	Purity (%)	92.2	91.8	90.9
	minDCF	**0.122**	0.124	**0.122**
K-Means	Purity (%)	84.2	86.8	85.4
	minDCF	0.237	0.226	0.250
K-Means + GMM	Purity	88.7	90.2	90.2
	minDCF	0.211	0.196	0.210

5.2. Effect of Cluster Purity

Table 1 reports the purity of each of the clustering techniques with regards to MFCC, PLP, and RASTA-PLP features. We use 13-dimensional acoustic features for segmentation and HAC clustering, while we additionally use their first and second derivatives (i.e. 39-dimensional features) for K-means and GMM clustering. This ensures high accuracy in detecting short-duration segments[3]. To assess the correlation between cluster purity and SAD accuracy, we report the SAD results obtained when applying a MFCC-based I-Vector + PLDA system on top of the clustered data.

It is worth noting that temporal smoothing on speech segments was not applied in this experiment in order to assess the discrimination power of the raw SAD scores.

Table 1 shows that GLR/BIC segmentation achieves the highest purity, while this segmentation followed by BIC hierarchical clustering achieves the highest accuracy with very competitive purity. K-means gets the worst results in terms of purity and accuracy, but following K-means with GMM clustering improves both the purity and the accuracy across all features.

Furthermore, Table 1 shows that MFCC produces the best results for GLR/BIC segmentation + BIC hierarchical clustering, while PLP produces the best results for K-means + GMM clustering. In the remaining experiments, we will apply the clustering techniques using their best feature matches.

5.3. Accuracy Without Smoothing

Table 2 summarizes the accuracy of SAD systems under the different data structuring techniques. Both overall and channel-specific minDCF values are reported. Similarly to the previous experiment, the acoustic features used in the classifier are MFCC and the temporal smoothing is not applied. Table 2 shows not only that segmentation followed by HAC clustering produces the highest accuracies, but also that the SAD system based on I-Vectors and SVM outperforms its competitors under this data structuring with an overall minDCF of 0.115. It also shows that all proposed systems outperform the frame-based GMM baseline, which achieves an overall minDCF of 0.242. Interestingly, Table 2 also shows that channel F is among the most difficult channels, with the best system achieving a

[3]The computation of determinants in Eq. 5 requires that the minimum number of feature vectors necessary to model a Gaussian distribution to be strictly greater than the feature dimension.

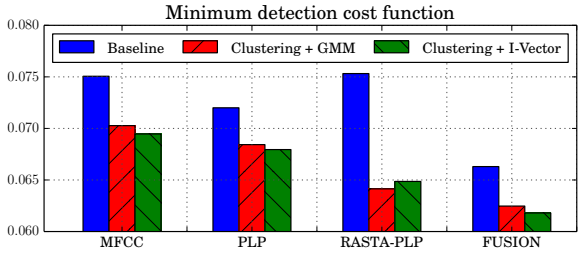

Figure 3: Performance of the baseline and proposed systems using MFCC, PLP, and RASTA-PLP. The clustering is K-Means + GMM.

minDCF of 0.141. This is because channel F contains a low-frequency noise component with very high energy that is hard to distinguish from the speech signal.

5.4. Accuracy After Smoothing

Table 3 reports the accuracy of the same SAD systems presented in Table 2, but after smoothing. Table 3 clearly shows that the accuracy of all SAD systems increased. The gain ranges from 40% to 222% depending on the system.

For the frame-based baseline system, minDCF dropped considerably from 0.242 to 0.075. K-Means + GMM clustering achieved the best overall performance, with minDCF for the I-Vector + SVM classifier dropping to 0.069.

It is worth noting that SVM performed better than PLDA. This is most likely because SAD is a binary classification task. Interestingly, the best SAD system without smoothing (Segmentation + HAC) improved by only 55% with smoothing (from 0.115 to 0.074), which was not enough to outperform the smoothed K-means + GMM clustering. The K-means + GMM clustering will be used in the following experiment.

5.5. Accuracy Across Features

Fig. 3 compares the proposed cluster-based I-Vector and GMM systems against the frame-based GMM baseline across MFCC, PLP, and RASTA-PLP features. It also presents the results of the score-level fusion for the three systems.

Fig. 3 clearly shows that cluster-based SAD systems outperform the baseline for all features. The relative improvement is 8%, 6%, 17% and 7% for MFCC, PLP, RASTA-PLP and the fusion, respectively. Results in Fig. 3 suggest that RASTA-PLP is more suitable for cluster-based SAD, while PLP is more suitable for frame-based SAD. Additionally, Fig. 3 shows that the I-Vector system is superior to GMM, except for RASTA-PLP features[4].

6. Conclusions

This paper introduces I-Vectors for speech activity detection. This is facilitated by first clustering the data, and then applying I-Vectors at the cluster level. Experimental results on the challenging RATS dataset in the context of the 2015 NIST OpenSAD evaluation show that the proposed approach outperforms the baseline frame-based GMM by up to 17% of relative improvement. Future work will focus on evaluating the impact of the proposed SAD technique on speaker recognition.

[4]Further analysis of per-channel performance for RASTA-PLP features reveals that I-Vector outperforms GMM for all channels except the difficult channel F.

7. Acknowledgements

We would like to thank Gregory A. Sanders from NIST for organizing openSAD and evaluating our system, and Ilya Ahtaridis from LDC for sharing with us the RATS corpus.

8. References

[1] M. J. Alam, P. Kenny, P. Ouellet, T. Stafylakis, and P. Dumouchel, "Supervised/unsupervised voice activity detectors for text-dependent speaker recognition on the rsr2015 corpus," in *Odyssey*, 2014, pp. 123–130.

[2] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[3] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 19, pp. 788–798, 2011.

[4] G. Saon, S. Thomas, H. Soltau, S. Ganapathy, and B. Kingsbury, "The ibm speech activity detection system for the darpa rats program.," in *INTERSPEECH*, 2013, pp. 3497–3501.

[5] N. Ryant, M. Liberman, and J. Yuan, "Speech activity detection on youtube using deep neural networks.," in *INTERSPEECH*. 2013, pp. 728–731, ISCA.

[6] S. Prince and J. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *IEEE ICCV*, 2007, vol. 0, pp. 1–8.

[7] Vladimir N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag New York, Inc., 1995.

[8] D. Graff, K. Walker, S. Strassel, X. Ma, K. Jones, and A. Sawyer, "The rats collection: Supporting HLT research with degraded audio data," in *LREC*. May 2014, pp. 1970–1977, European Language Resources Association (ELRA).

[9] A. Benyassine, E. Shlomot, H. Su, D. Massaloux, C. Lamblin, and J.-P. Petit, "ITU-T Recommendation G.729 Annex B: a silence compression scheme for use with G.729 optimized for V.70 digital simultaneous voice and data applications," *IEEE Communications Magazine*, vol. 35, no. 9, pp. 64–73, 1997.

[10] T. Kinnunen and P. Rajan, "A practical, self-adaptive voice activity detector for speaker verification with noisy telephone and microphone data," in *IEEE ICASSP*, 2013.

[11] E. Scheirer and M. Slaney, "Construction and evaluation of a robust multifeature speech/music discriminator," in *IEEE ICASSP*, 1997, vol. 2, pp. 1331–1334.

[12] Z.-H. Tan and B. Lindberg, "Low-complexity variable frame rate analysis for speech recognition and voice activity detection," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 5, pp. 798–807, 2010.

[13] D. Ying, Y. Yan, J. Dang, and F.K. Soong, "Voice activity detection based on an unsupervised learning framework," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, no. 8, pp. 2624–2633, 2011.

[14] T. Hain and P. C. Woodland, "Segmentation and classification of broadcast news audio," in *Proceedings of International Conference on Spoken Language Processing (ICSLP 98)*, Sydney, Australia, November 1998.

Table 2: Performance summary of SAD systems with regards to different data structuring techniques **without temporal smoothing**. The features used for SAD are MFCCs.

Data structuring	SAD classifier	Overall	B	D	E	F	G	H
Frame-based (Baseline)	GMM	0.242	0.287	0.254	0.274	0.194	0.193	0.265
Segmentation	GMM	0.131	0.132	0.131	0.101	0.185	0.119	0.105
	I-Vector + PLDA	0.131	0.141	0.135	0.105	0.158	0.124	0.121
	I-Vector + SVM	0.120	0.130	0.125	0.098	0.149	**0.106**	0.108
Segmentation + HAC	GMM	0.133	0.130	0.133	0.103	0.197	0.118	0.107
	I-Vector + PLDA	0.122	0.132	0.124	0.093	0.157	0.112	0.110
	I-Vector + SVM	**0.115**	**0.118**	**0.123**	**0.091**	0.144	**0.106**	**0.101**
K-Means	GMM	0.199	0.269	0.251	0.130	0.187	0.191	0.153
	I-Vector + PLDA	0.226	0.257	0.247	0.205	0.201	0.230	0.210
	I-Vector + SVM	0.226	0.251	0.244	0.216	0.206	0.217	0.225
K-Means + GMM	GMM	0.150	0.201	0.182	0.105	**0.141**	0.145	0.117
	I-Vector + PLDA	0.196	0.230	0.199	0.191	0.169	0.199	0.190
	I-Vector + SVM	0.200	0.234	0.197	0.198	0.175	0.193	0.211

Table 3: Performance summary of SAD systems with regards to different data structuring techniques **after temporal smoothing**. The features used for SAD are MFCCs.

Data structuring	SAD classifier	Overall	B	D	E	F	G	H
Frame-based (Baseline)	GMM	0.075	0.083	0.060	0.068	0.114	0.056	0.067
Segmentation	GMM	0.090	0.085	0.065	0.072	0.176	0.065	0.071
	I-Vector + PLDA	0.093	0.085	0.068	0.066	0.141	0.064	0.078
	I-Vector + SVM	0.085	0.075	0.064	0.074	0.146	0.065	0.082
Segmentation + HAC	GMM	0.093	0.084	0.066	0.071	0.191	0.064	0.072
	I-Vector + PLDA	0.083	0.079	0.068	0.079	0.177	0.068	0.076
	I-Vector + SVM	0.074	0.068	0.058	0.062	0.133	0.055	0.066
K-Means	GMM	0.081	0.072	0.062	0.077	0.151	0.054	**0.063**
	I-Vector + PLDA	0.113	0.155	0.106	0.089	0.155	0.081	0.089
	I-Vector + SVM	0.084	0.101	0.078	0.061	0.144	0.053	**0.063**
K-Means + GMM	GMM	0.070	**0.064**	0.053	0.074	**0.113**	0.052	**0.063**
	I-Vector +PLDA	0.072	0.077	0.060	**0.057**	0.119	0.051	**0.063**
	I-Vector +SVM	**0.069**	0.066	**0.052**	**0.057**	0.128	**0.048**	**0.063**

[15] L. Ferrer, M. McLaren, N. Scheffer, Y. Lei, M. Graciarena, and V. Mitra, "A noise-robust system for nist 2012 speaker recognition evaluation," in *INTERSPEECH*, 2013.

[16] M. McLaren, M. Graciarena, and Y. Lei, "Softsad: Integrated frame-based speech confidence for speaker recognition," in *IEEE ICASSP*, 2015, pp. 4694–4698.

[17] T. Hughes and K. Mierle, "Recurrent neural networks for voice activity detection," in *IEEE ICASSP*, 2013, pp. 7378–7382.

[18] F. Eyben, F. Weninger, S. Squartini, and B. Schuller, "Real-life voice activity detection with lstm recurrent neural networks and an application to hollywood movies," in *IEEE ICASSP*, 2013, pp. 483–487.

[19] S. Thomas, G. Saon, M. Van Segbroeck, and S. Narayanan, "Improvements to the ibm speech activity detection system for the darpa rats program," in *IEEE ICASSP*, 2015.

[20] H. Gish, M.H. Siu, and R. Rohlicek, "Segregation of speakers for speech recognition and speaker identification," in *IEEE ICASSP*, 1991, pp. 873–876.

[21] S. Chen and P. Gopalakrishnan, "Clustering via the bayesian information criterion with applications in speech recognition," in *IEEE ICASSP*, 1998, vol. 2, pp. 645–648.

[22] E. Khoury, C. Senac, and J. Pinquier, "Improved speaker diarization system for meetings," *IEEE ICASSP*, pp. 4097–4100, 2009.

[23] X. Anguera Miro, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 2, pp. 356–370, 2012.

[24] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures.," *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[25] J. Rissanen, *Stochastic Complexity in Statistical Inquiry Theory*, World Scientific Publishing Co., Inc., River Edge, NJ, USA, 1989.

[26] C. Barras, X. Zhu, S. Meignier, and J.-L. Gauvain, "Improving speaker diarization," in *Proc. of DARPA RT04*, Palisades, USA, 2004.

[27] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, series B*, vol. 39, no. 1, pp. 1–38, 1977.

[28] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *INTERSPEECH*, 2011, pp. 249–252.

[29] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*. 1999, pp. 61–74, MIT Press.

[30] S. Pigeon, P. Druyts, and P. Verlinde, "Applying logistic regression to the fusion of the NIST'99 1-speaker submissions," *Digital Signal Processing*, vol. 10, no. 1–3, pp. 237–248, 2000.