Interspeech 2018
2-6 September 2018, Hyderabad

ZCU-NTIS Speaker Diarization System for the DIHARD 2018 Challenge

Zbyněk Zajíc[1], *Marie Kunešová*[1,2], *Jan Zelinka*[1,2], *Marek Hrúz*[1]

University of West Bohemia
Faculty of Applied Sciences
[1]NTIS - New Technologies for the Information Society and [2]Dept. of Cybernetics,
Univerzitní 8, 306 14 Plzeň, Czech Republic
{zzajic,mkunes,zelinka,mhruz}@ntis.zcu.cz

Abstract

In this paper, we present the system developed by the team from the New Technologies for the Information Society (NTIS) research center of the University of West Bohemia, for the First DIHARD Speech Diarization Challenge. The base of our system follows the currently-standard approach of segmentation, i-vector extraction, clustering, and resegmentation. Here, we describe the modifications to the system which allowed us to apply it to data from a range of different domains. The main contribution to our achievement is a Neural Network (NN) based domain classifier, which categorizes each conversation into one of the ten domains present in the development set. This classification determines the specific system configuration, such as the expected number of speakers and the stopping criterion for the hierarchical clustering. At the time of writing of this abstract, our best submission achieves a DER of 26.90% and an MI of 8.34 bits on the evaluation set (gold speech/nonspeech segmentation).

Index Terms: speaker diarization, speaker change detection, i-vector, statistics accumulation, agglomerative hierarchical clustering, neural network classifier

1. Introduction

In this paper, we present our off-line Speaker Diarization (SD) system [1, 2, 3] that was applied in the First DIHARD Speech Diarization Challenge [4]. This system has been used previously primarily for telephone data. The DIHARD Challenge brought an opportunity to apply our approach to a more diverse set of data. However, this required certain modifications of our system, which we describe in this paper. The most important new enhancement of our system is an application of an NN-based domain classifier that allows the system to automatically identify the domain of each recording and to set the system's configuration accordingly. This proved to be essential during the DIHARD Challenge, as the challenge data consist of multiple corpora with very different characteristics. Other modifications include a different clustering process (agglomerative rather than k-means) and a new speech activity detector.

The paper is organized as follows. Section 2 describes the main components of our system. Section 3 introduces the domain classifier. Section 4 describes the speech activity detector. Section 5 introduces the adult-child classifier. Finally, section 6 gives the results on the development and evaluation data.

2. Speaker Diarization System

Our system follows an i-vector-based approach, as introduced in [5, 6, 7]: First, each recording is divided into short segments and i-vectors are extracted. Then, a clustering method is used in order to determine which parts of the signal were produced by the same speaker. Finally, a GMM-based resegmentation is performed to refine the positions of boundaries between speakers.

For the DIHARD Challenge, we have also introduced a domain classifier that determines the source of each recording and selects the most suitable system configuration. A diagram of our diarization system is shown in Figure 1.

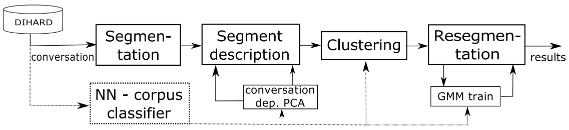

Figure 1: *Diagram of the diarization process.*

This section provides a description of the main steps of the diarization process. The domain classifier and related domain-dependent settings are described in section 3. The data we used for training each part of the system are listed in section 6.1.

2.1. Feature Extraction

We used Linear Frequency Cepstral Coefficients (LFCCs), Hamming window of length 25 ms with 10 ms shift. There are 40 triangular filter banks linearly spread across the frequency spectrum, and 25 LFCCs are extracted. The resultant 50-dimensional feature vector ($D_f = 50$) also includes delta coefficients.

2.2. Segmentation

In the segmentation step, each recording is split into multiple individual speech regions by breaking it on any non-speech longer than 0.5 s. Then, one of the following two methods is applied to further divide each long segment.

2.2.1. Fixed Length Segmentation

We simply cut the long speech regions at regular intervals, into segments with a length of 2 s and with a 1 s overlap between neighboring segments. If the remainder is shorter than 1 s, we extend it to 1 s by adding frames from the preceding segment.

2.2.2. CNN-based Speaker Change Detection

Our second segmentation approach uses a Convolutional Neural Network (CNN) to detect speaker changes [2]. The CNN was trained as a regressor on spectrograms of acoustic signal, with reference information L about existing speaker changes. The

10.21437/Interspeech.2018-1252

output signal P gives the probability of a speaker change at each given moment (see Figure 2).

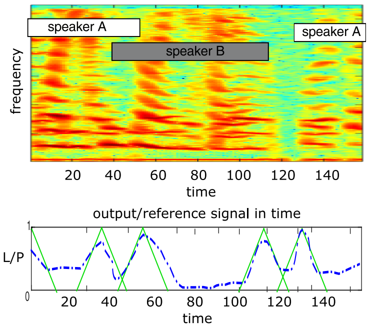

Figure 2: *The input speech, given as a spectrogram, is processed by the CNN into the output function P (probability of a speaker change at the given time). The lower image shows the reference function L and the output signal P.*

Speaker changes are identified as peaks in the signal P (after normalization into interval $< 0, 1 >$ for each conversation), using non-maximum suppression with a window size of 10 features. We also apply a threshold of 0.5 on the detected peaks in order to remove insignificant local maxima. The signal between two detected speaker changes is considered as one segment.

To ensure that each segment contains sufficient information about the speaker, we set the minimum duration of each segment at one second. Shorter segments are discarded and the decision about the speaker is left for the resegmentation step.

The network was only trained on one part of the development set of the DIHARD corpus [8], specifically the YouthPoint radio interviews, because they appear to contain the fewest transcription errors.

2.3. Segment Description

Each segment is represented by an i-vector derived from the supervector of accumulated statistics [9] - zeroth and first statistical moments of data related to a UBM as a GMM with $M = 1024$ components. The dimensionality of this supervector is reduced by Factor Analysis (FA) [10] into $D_w = 100$ (details about the training of the total variability space matrix can be found in [11, 12]) and we have used conversation-dependent Principal Component Analysis (PCA) [7] to reduce the dimension further into 3 or 9 (depending on the specific data - see Tab. 1).

2.3.1. Statistics Refinement

Because we cannot be certain that each segment only contains the speech of a single speaker, not all data from a segment should contribute to the supervector equally. With CNN-based segmentation, we can reuse the output of the CNN (the probability of a speaker change in the signal) as an indication of the suitability of each frame. The part of the audio segment in time t with a high probability of a speaker change $P(t)$ is less appropriate to represent the speaker than a part with a small value of $P(t)$. Thus, we use the value of $1 - P(t)$ as a weighting factor of the signal during the accumulation process [3]. However, no such weighing is applied with fixed length segmentation.

2.4. Clustering

In the DIHARD corpus, the number of speakers in each recording is unknown in advance. In the development set, it varies between 1 and 10 speakers, depending on the domain. Thus, we have chosen to primarily use the agglomerative hierarchical clustering (AHC) algorithm. However, we make an exception for two specific corpora where we are reasonably certain of the number of speakers. For them, we use k-means instead.

2.4.1. Agglomerative Clustering

The system starts with each i-vector in a separate cluster and then merges the closest pairs until it reaches a stopping point. The distance between two clusters is calculated as the average cosine distance[1]between each pair of i-vectors. The stopping condition is a combination of maximum merging distance and a minimum and a maximum number of clusters:

First, we perform AHC by merging the closest pairs of clusters until the lowest distance exceeds a specific threshold. If the resulting number of clusters is not within the expected range, we adjust the stopping point so that we reach either the minimum or maximum allowed number of clusters.

These parameters were selected on a per-corpus basis using the development set. The target number of speakers is based on the actual numbers in each conversation, while the optimal threshold for the merging distance was found experimentally (see section 6).

2.4.2. K-means Clustering

While the number of speakers in most of the DIHARD recordings varies even within each domain, two of the corpora in the development set almost exclusively contain exactly two speakers in each conversation. For these two domains, we simply applied k-means clustering into 2 clusters, using cosine distance between i-vectors.

2.5. Resegmentation

To make the final diarization more precise, we refine it by resegmentation. We compute GMMs over the feature vectors, one GMM for each speaker cluster. Then the whole conversation is redistributed frame by frame according to the likelihoods of the GMMs, filtered by a Gaussian window (length 75 ms with shift 50 ms) to smooth the peaks in the likelihoods. The number of GMM components depends on the amount of data in each cluster and ranges between 1 and 64.

3. Domain Classification

The DIHARD corpus [8] consists of data taken from several different domains, with very diverse characteristics - including the number of speakers, the level of noise and general audio quality. As such, it is difficult to find a single system configuration which would work well for the entirety of the data.

To resolve this issue, we have implemented a domain classifier - a neural network which receives a single i-vector calculated over the entire conversation and outputs the probability of each of the 9 corpora in the DIHARD development set.

The network was implemented in TensorFlow[2]. It was trained with one hidden layer (2048 neurons, tanh activation

[1] We have also investigated the use of a PLDA model [13, 14] for calculating the similarity of two i-vectors, but it did not bring any improvements.

[2] https://www.tensorflow.org

function) followed by 0.9 dropout and the output layer as softmax into 9 categories.

As the evaluation set contains additional unseen corpora, we have also added a threshold (= 0.5) on the output probability from the classifier and categorize lower-scoring conversations as “unknown domain”. The accuracy of the trained classifier was 95% on the development set.

4. Speech Activity Detection

The DIHARD Challenge consisted of two tracks - diarization using gold speech segmentation, and diarization from scratch. While our main efforts focused on the first track, we have also submitted a system for the second track. This section describes the speech activity detector we used for the purpose.

Our SAD system is an enhanced version of the approach described in [15].

We use a neural network (depicted in Figure 3) consisting of three parts: The first part computes a spectral flux or other analogous spectral features and also contains a feature extractor. The second part computes a score from contextual information. And the last part functions as a decoder.

The first part is three standard CNN layers with ReLU activation function, each followed by a pooling layer with max pooling function, and ending with four standard fully-connected NN layers with sigmoid activation functions. The second part is a splicing that makes a long-temporal window and one single neuron with linear activation function. The last part also uses a long-temporal window and it finds the maximal score in the window.

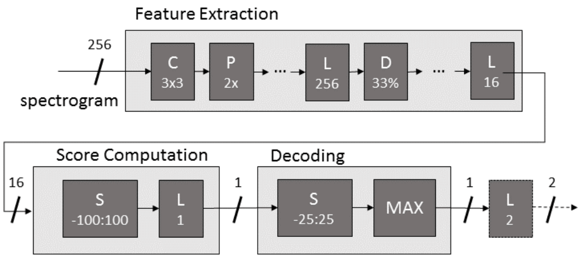

Figure 3: *The schema of the NN for SAD, where C = convolution, P = pooling, D = dropout, L = fully-connected and S = splicing layer.*

Our SAD processes the logarithm of amplitude spectrum that is computed from 512 samples long window with 160 samples step (hop). Thus, the spectrum has 256 features. Each CNN layer uses a 3x3 window and it has 6 kernels. Fully-connected layers in the second part have 256 neurons, except for the last layer, which has 16 neurons. The window splices 201 feature vectors. The window is symmetric, i.e. used time shifts are from -100 to +100. The window in the third process is also symmetric and the length of the window is 51.

In the training process, we added one layer with softmax activation function and we trained all parts simultaneously by means of cross-entropy criterion and SGD. Due to the long windows, we used batches of 128 randomly selected continuous parts of recordings, always 2000 spectral features long.

5. Adult-Child classification

During our experiments on the development set, we found that the AHC approach described in section 2 performed very poorly on data containing the speech of adults and very young children (i.e. the SEEDLingS corpus [16]), to the extent where we achieved the lowest DER when assigning all speech to the same speaker. For this reason, we have tried to use a different approach for this particular domain.

We used the following simple approach: Only two speakers are expected in the recording - one child and one adult. We have prepared a separate UBM for children and for adults and we classify each frame of the recordings as one of the two categories, using the same algorithm we use for resegmentation. After this adult-child classification, we also use a regular resegmentation using GMMs created from the specific conversation, as described in section 2.5.

6. Experiments

This section describes our experiments on the development set of the DIHARD Challenge, as well as our final results on the evaluation set. The experiments mainly served for finding the optimal system configuration for each of the individual corpora. For details of the DIHARD corpus [8, 16], see the evaluation plan [4].

6.1. Training Data

This section gives the complete list of the data we used for training each part of the system.

The *speech activity detector* was trained only on the DIHARD development set. For the *domain classifier*, we also added 10 recordings from the LibriSpeech[3] corpus as additional LibriVox data.

CNN-based segmentation: The CNN was trained only on the YouthPoint subset of the DIHARD development data.

i-Vector extraction: The UBM was trained on subsets of LibriSpeech, AMI Corpus[4], and the following ELRA and LDC corpora: Speecon database (Child voices only) - Czech (ELRA-S0298), UK English (ELRA-S0215) and US English (ELRA-S0233), TIMIT Acoustic-Phonetic Continuous Speech Corpus (LDC93S1), CSR-I (WSJ0) Complete (LDC93S6A), CSR-II (WSJ1) Complete (LDC94S13A), RT-03 MDE Training Data Speech (LDC2004S08), Santa Barbara Corpus of Spoken American English Part II (LDC2003S06).

Child-adult classifier: The adult GMM was trained on the same data as the general UBM for i-vector extraction (listed above), with the exception of the Speecon child voices, which were instead used to train the child UBM (150 children aged 8-15). Both these models were further adapted on the SEEDLingS development data, which were manually divided into clean child and adult segments. For the child UBM, we also obtained several other short recordings of small children (6 children between 3 months and 3 years old, total length 6 min), which were added to the adaptation data. Note: To avoid over-training on the development data, the adaptation of both UBM models on SEEDLingS data was only used for diarization on the evaluation set.

[3] http://www.openslr.org/12/

[4] http://groups.inf.ed.ac.uk/ami/download/

6.2. Evaluation

The system performance was evaluated in terms of Diarization Error Rate (DER), as defined by NIST [17]. On the development set, we calculated this on a per-recording basis using NIST's md-eval-v21.pl script[5]. Results on the evaluation set were given by the official scoring system.

Unlike usual practice, DIHARD Challenge submissions were scored with no forgiveness collar around speaker boundaries, and overlapping speech was included in the evaluation.

6.3. Domain-specific settings

Because of our domain classifier, we were able to use different system configurations for each of the nine development set corpora and for unknown data. Here we describe the general approaches we selected for each domain. Specific experimentally-chosen parameters are listed in Table 1.

SCOTUS, YouthPoint, SLX and RT-04S: For these corpora, we used the AHC approach, as described in section 2.4.

ADOS and DCIEM: Both corpora had almost exclusively exactly 2 speakers in each conversation. For this reason, we could simply use k-means clustering with 2 clusters.

LibriVox: All recordings contained only 1 speaker. Thus, we did not need to perform diarization, but simply used the information given by SAD or gold segmentation.

VAST: On this corpus, our system did not work well - it achieved the lowest DER when all speech was assigned to a single cluster. Thus, we simply used the same method as with LibriVox data.

SEEDLingS: This corpus had the same issue as VAST. As an alternate solution, we applied the child-adult classifier described in section 5. This improved the SEEDLings DER on the development set from 36.24% to 29.50%, but on the evaluation set the difference was negligible.

Unknown: For unrecognized evaluation data, we've chosen to use AHC with 3 target clusters.

Table 1: *Experimentally chosen parameters (Thr. = threshold, k-m = k-means, A/Ch = adult/Child segmentation) for each corpus and segmentation approaches (fixed length win. or CNN).*

corpus	Thr. SAD	Clus-tering	No. spk	Thr. AHC fix. len./ CNN	PCA dim
SEEDL.	0.60	A/Ch	2	-	-
SCOTUS	0.95	AHC	5-10	0.64	9
DCIEM	1.15	k-m	2	-	3
ADOS	1.10	k-m	2	-	3
YouthP.	0.90	AHC	3-5	0.64/0.62	9
SLX	1.10	AHC	2-6	0.74/0.58	9
RT-04S	-0.55	AHC	3-10	0.60/0.76	9
LibriVox	1.00	-	1	-	-
VAST	0.55	-	1	-	-
other	0.80	AHC	3	-	9

6.4. Results

Table 2 shows system results on the development set for each of the nine corpora, with both types of segmentation. Table 3 then presents the final results on the evaluation data for both tracks - diarization from gold segmentation (Track 1) and diarization from scratch (Track 2).

Table 2: *Average DER [%] on individual corpora of the DIHARD development set, for a system with fixed length segmentation and a system with CNN-based speaker change detection (both with resegmentation).*

system	fixed length	CNN-SCD
SEEDLingS	29.50	29.50
SCOTUS	8.27	9.03
DCIEM	10.01	10.07
ADOS	17.00	16.95
YouthPoint	3.81	4.30
SLX	21.92	26.63
RT-04S	36.25	40.89
LibriVox	0.00	0.00
VAST	32.38	33.32
All	19.93	20.95

Table 3: *Official results (DER [%] and MI [bits]) on the DIHARD evaluation data for both types of segmentation.*

system	track1	track2
Fixed length seg.	DER: 26.90%, MI: 8.34 bits	DER: 45.78%, MI: 7.79 bits
CNN-SCD	DER:27.12%, MI: 8.31 bits	DER: 46.14%, MI: 7.77 bits

7. Conclusion

In this paper, we presented a new version of our diarization system, which was created for the DIHARD Diarization Challenge. Using a domain classifier, we were able to use a different system configuration for each subset of the challenge data.

Our system performed well on relatively clean data (YouthPoint, SCOTUS, DCIEM). Other corpora proved more challenging. However, due to a limited amount of time and personal capacity, we were not able to focus on all possible areas of improvement - such as the detection of crosstalk and environmental noise, or better tuning of our Track 2 submissions. The approach we used for the SEEDLingS child speech (section5) is also rather simplistic and in need of further improvement.

Despite our system's limitations, it has placed reasonably well in the DIHARD Challenge. Our best Track 1 submission achieved a DER of 26.90% and at the time of this writing, ranks in the fifth place of the 14 teams on the leaderboard (by DER).

8. Acknowledgements

This research was supported by the Grant Agency of the Czech Republic, project No. GAČR GBP103/12/G084. Acknowledgements also to the Ministry of Education, Youth and Sports of the Czech Republic project No. LO1506 that provided the computational infrastructure to achieve the presented results and supported by the grant of the University of West Bohemia, project No. SGS-2016-039. The authors would also like to thank their colleagues who provided recordings of their children as additional training data.

[5] https://web.archive.org/web/20160110193918/http://www.itl.nist.gov:80/iad/mig/tests/rt/2006-spring/code/md-eval-v21.pl

9. References

[1] Z. Zajíc, M. Kunešová, and V. Radová, “Investigation of Segmentation in i-Vector Based Speaker Diarization of Telephone Speech,” in *Specom*. Budapest: Springer, 2016, pp. 411–418.

[2] M. Hrúz and Z. Zajíc, “Convolutional Neural Network for Speaker Change Detection in Telephone Speaker Diarization System,” in *ICASSP*. New Orleans: IEEE, 2017, pp. 4945–4949.

[3] Z. Zajíc, M. Hrúz, and L. Müller, “Speaker diarization using convolutional neural network for statistics accumulation refinement,” in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, Stockholm, 2017, pp. 3562–3566.

[4] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, “First DIHARD Challenge Evaluation Plan,” Tech. Rep., 2018. [Online]. Available: https://zenodo.org/record/1199638

[5] G. Sell and D. Garcia-Romero, “Speaker Diarization with PLDA I-vector Scoring and Unsupervised Calibration,” in *IEEE Spoken Language Technology Workshop*, South Lake Tahoe, 2014, pp. 413–417.

[6] M. Senoussaoui, P. Kenny, T. Stafylakis, and P. Dumouchel, “A Study of the Cosine Distance-Based Mean Shift for Telephone Speech Diarization,” *Audio, Speech and Language Processing*, vol. 22, no. 1, pp. 217–227, 2014.

[7] S. Shum, N. Dehak, E. Chuangsuwanich, D. Reynolds, and J. Glass, “Exploiting Intra-Conversation Variability for Speaker Diarization,” in *Interspeech*, Florence, 2011, pp. 945–948.

[8] N. Ryant *et al.*, “DIHARD Corpus.” Linguistic Data Consortium, 2018.

[9] Z. Zajíc, L. Machlica, and L. Müller, “Robust Statistic Estimates for Adaptation in the Task of Speech Recognition,” in *TSD*, vol. 6231. Brno: Springer, 2010, pp. 464–471.

[10] P. Kenny and P. Dumouchel, “Experiments in Speaker Verification Using Factor Analysis Likelihood Ratios,” in *Odyssey*, Toledo, 2004, pp. 219–226.

[11] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, “A Study of Interspeaker Variability in Speaker Verification,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 5, pp. 980–988, 2008.

[12] L. Machlica and Z. Zajíc, “Factor Analysis and Nuisance Attribute Projection Revisited,” in *Interspeech*, vol. 2, Portland, 2012, pp. 1570–1573.

[13] S. J. Prince and J. H. Elder, “Probabilistic Linear Discriminant Analysis for Inferences About Identity,” in *Computer Vision*. Rio de Janeiro: IEEE, 2007, pp. 1–8.

[14] L. Machlica and Z. Zajíc, “An efficient implementation of Probabilistic Linear Discriminant Analysis,” in *ICASSP*. Vancouver: IEEE, 2013, pp. 7678–7682.

[15] S. Thomas, S. Ganapathy, G. Saon, and H. Soltau, “Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions,” in *ICASSP*. Florence: IEEE, may 2014, pp. 2519–2523.

[16] E. Bergelson, “Bergelson Seedlings HomeBank Corpus,” 2018, doi:10.21415/T5PK6D.

[17] J. G. Fiscus, N. Radde, J. S. Garofolo, A. Le, J. Ajot, and C. Laprun, “The Rich Transcription 2006 Spring Meeting Recognition Evaluation,” *Machine Learning for Multimodal Interaction*, vol. 4299, pp. 309–322, 2006.