INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Speaker Diarization Using Convolutional Neural Network for Statistics Accumulation Refinement

Zbyněk Zajíc[1], *Marek Hrúz*[1], *Luděk Müller*[1,2]

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@ntis.zcu.cz, mhruz@ntis.zcu.cz, muller@ntis.zcu.cz

Abstract

The aim of this paper is to investigate the benefit of information from a speaker change detection system based on Convolutional Neural Network (CNN) when applied to the process of accumulation of statistics for an i-vector generation. The investigation is carried out on the problem of diarization. In our system, the output of the CNN is a probability value of a speaker change in a conversation for a given time segment. According to this probability, we cut the conversation into short segments that are then represented by the i-vector (to describe a speaker in it). We propose a technique to utilize the information from the CNN for the weighting of the acoustic data in a segment to refine the statistics accumulation process. This technique enables us to represent the speaker better in the final i-vector. The experiments on the English part of the CallHome corpus show that our proposed refinement of the statistics accumulation is beneficial with the relative improvement of Diarization Error Rate almost by 16 % when compared to the speaker diarization system without statistics refinement.

Index Terms: convolutional neural network, speaker change detection, speaker diarization, i-vector, statistics accumulation

1. Introduction

The problem of Speaker Diarization (SD) is crucial for many speech applications dealing with real data, where only one speaker occurrence in a recording cannot be ensured. The SD problem is defined as a task of categorizing speakers in an unlabeled conversation, without any prior information regarding the number and identities of the speakers. Different approaches were proposed to solve this task [1]. The most common approach to the SD consists of the segmentation of an input signal, followed by the merging of the segments into clusters corresponding to individual speakers [2, 3]. Alternatively, the segmentation and the clustering step can be combined into a single iterative process [4]. In this paper, we investigate the state-of-the-art off-line SD system based on the i-vector representation of the speech segments [3, 5] (other approaches utilize e.g. Hidden Markov Models [6, 7]).

The speaker change detection (SCD) is often applied to the audio signal to obtain segments which ideally contain a speech of a single speaker [2]. Commonly used approaches to the SCD include the Bayesian Information Criterion (BIC), Generalized Likelihood Ratio (GLR), Kullback-Leibler divergence [8, 9], Support Vector Machine (SVM) [10] and Deep Neural Networks (DNNs) [11, 12]. However, in a spontaneous telephone conversation containing very short speaker turns and frequent overlapping speech, diarization systems often omit the SCD process and use a simple constant length window segmentation of speech [3, 5].

The success of DNNs in the speech recognition task [13] leads in recent times to their exploitation in SD systems. DNNs are utilized in the task of the segmentation [11, 14] or in the clustering process [15, 16]. In [17] DNNs are used to replace unsupervised Universal Background Model (UBM) for the accumulation of statistics in the i-vector generation. DNN was also applied to the representation of the speaker in [18, 19] or very recently in [20] and in [21], where the triplet loss paradigm was used for training the DNN descriptor with extremely short speech turn.

In our previous papers [14, 22] we applied a CNN to the problem of SCD. The main difference between our approach and the one in others works lies in the fact that we introduce a spectrogram to a CNN and let the net compute its own features.

CNNs were introduced in [23] to cope with the problem of image classification. They were popularized by Krizhevsky *et al.* [24] with updated design blocks such as Rectified Linear Units (ReLU) or max pooling instead of average pooling. When a CNN is trained on large scale datasets one can observe its capability to learn discriminative features on its own. Furthermore, the net is able to learn a semantic representation of the data. Our experiments with the CNN in the task of SCD exhibited better results than classical approaches based on BIC. The input of the network is a spectrogram of a segment of the original waveform and the output is a probability that there is a speaker change in the middle of the segment. When the CNN is applied to the whole recording in a sliding window fashion a probability signal of the speaker change is obtained. Further processing of this signal is needed to determine where a change occurs. In our previous work, we detected peaks using non-maximum suppression.

In this paper, our goal is to determine whether the CNN also offers any useful information about the homogeneity of a speaker in a segment. For this purpose, we propose a refinement of accumulation of statistics for i-vector generation and apply it to our SD system [14].

2. Speaker Diarization System

Our SD system [14] is based on the i-vectors [25] that represent speech segments, as introduced in [26]. These segments are obtained from the previous step using SCD based on CNN. The resulting i-vectors are clustered in order to determine which parts of the signal were produced by the same speaker. A diagram of our diarization system can be seen in Figure 1.

http://dx.doi.org/10.21437/Interspeech.2017-51

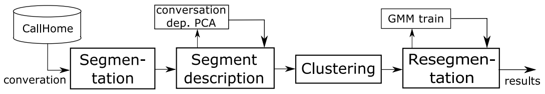

Figure 1: *Diagram of the diarization process.*

2.1. Segmentation

For the segmentation step, we use the SCD approach based on CNN [14]. The CNN as a regressor is trained supervised on spectrograms of the acoustic signal with a reference information L about the existing speaker changes. The value of the function L in time t is computed via the formula in Equation 1. We call this labeling a fuzzy labeling. It has a shape of a triangle and the main idea behind it is to model the uncertainty of human labeling.

$$L(t) = \max\left(0, 1 - \frac{\min_i\left(|t - s_i|\right)}{\tau}\right), \tag{1}$$

where s_i is the time of i^{th} speaker change and $\tau = 0.6$ is the tolerance which models the level of uncertainty of the manual labeling. Figure 2 depicts an example of a spectrogram, the values of the labeling and the CNN output as a probability of speaker change P (a number between zero and one). The speaker changes are identified as peaks in the signal P using non-maximum suppression with a suitable window size. The detected peaks are then thresholded to remove insignificant local maxima. The signal between two detected speaker changes is considered as one segment. The minimum duration of one segment is limited to one second, smaller segments are joined to the adjacent one in order to obtain sufficient information about the speaker.

2.2. Segment description

To describe a segment we first construct a supervector of accumulated statistics. Supervectors have been used in the process of speaker adaptation [27] where they serve as a descriptor of a new speaker. They contain the zeroth and first statistical moments of speakers' data related to a UBM. The UBM is modeled as a Gaussian Mixture Model (GMM) from a huge amount of speech data form different speakers. The parameters of the model are $\boldsymbol{\lambda}_{\text{UBM}} = \{\omega_m, \boldsymbol{\mu}_m, \boldsymbol{C}_m\}_{m=1}^{M}$, where M is the number of mixtures in the UBM, ω_m, $\boldsymbol{\mu}_m$, $\boldsymbol{C}_m$ are the weight, mean and covariance of the m^{th} mixture, respectively. We consider only diagonal covariance matrices.

Let $\boldsymbol{O} = \{\boldsymbol{o}_t\}_{t=1}^{T}$ be the set of T feature vectors $\boldsymbol{o}_t$ of a dimension D of one segment of conversation, and

$$\gamma_m(\boldsymbol{o}_t) = \frac{\omega_m \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{\mu}_m, \boldsymbol{C}_m)}{\sum_{m=1}^{M} \omega_m \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{\mu}_m, \boldsymbol{C}_m)} \tag{2}$$

be the posterior probability of m^{th} mixture given a feature vector $\boldsymbol{o}_t$. The soft count of the m^{th} mixture (zeroth statistical moment of feature vectors) is

$$n_m = \sum_{t=1}^{T} \gamma_m(\boldsymbol{o}_t) \tag{3}$$

and the sum of the first statistical moments of feature vectors with respect to the m^{th} mixture is

$$\boldsymbol{b}_m = \sum_{t=1}^{T} \gamma_m(\boldsymbol{o}_t)\boldsymbol{o}_t. \tag{4}$$

The speaker's supervector $\boldsymbol{\psi}$ [28] for given data $\boldsymbol{O}$ is a concatenation of the zeroth and first statistical moments of $\boldsymbol{O}$. Our proposed refinement of this process of statistics accumulation is described in Section 3.

Next, we extract the i-vectors from the supervectors. Supervectors have usually a high dimension $D = M * (D_f + 1)$ that is given by the number of mixtures M in the UBM and the D_f dimensionality of the feature vectors $\boldsymbol{o}_t$. The i-vectors are a compact representation of the information encoded in the supervectors, mostly the information about the identity of the speaker. Factor Analysis (FA) [29] (or extended Joint Factor Analysis (JFA) [30] to handle more sessions of each speaker) is used for dimensionality reduction of the supervector of statistics. The generative i-vector model has the form

$$\boldsymbol{\psi} = \boldsymbol{m}_0 + \boldsymbol{T}\boldsymbol{w} + \boldsymbol{\epsilon}, \;\; \boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \;\; \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}), \tag{5}$$

where $\boldsymbol{T}$ (of size $D \times D_w$) is called the total variability space matrix, $\boldsymbol{w}$ is the segment's i-vector of dimension D_w having standard Gaussian distribution, $\boldsymbol{m}_0$ is the mean vector of $\boldsymbol{\psi}$, however often approximated by the UBM's mean supervector, and $\boldsymbol{\epsilon}$ is residual noise with a diagonal covariance matrix $\boldsymbol{\Sigma}$ with covariance matrices $\boldsymbol{C}_1, \ldots, \boldsymbol{C}_M$ of the UBM ordered on the diagonal. The i-vectors are also length-normalized [31]. Details about the training of total variability space matrix $\boldsymbol{T}$ can be found in [32, 33].

Because of the differences between each conversation (and the similarity in one conversation), we also compute a conversation dependent Principal Component Analysis (PCA) transformation [26], which further reduces the dimensionality of the i-vector. The benefit of using PCA instead of FA approach is the additional information about the importance of each component given by the eigenvalue of the corresponding eigenvector. The reduced dimension in the PCA latent space can be found for each conversation separately depending only on the ratio of eigenvalue mass.

2.3. Clustering and Resegmentation

Given i-vector representations of the extracted segments, we perform a clustering into sets of i-vectors describing different speakers. This is a coarse clustering on the level of the segmentation given by SCD. To make the final diarization more precise we refine it by resegmentation. We compute GMMs over the feature vectors $\boldsymbol{o}_t$, one GMM per speaker cluster. Then the whole conversation is redistributed frame by frame according to the likelihoods of the GMMs.

3. Statistics Refinement

Because of the uncertainty about the assumption that there is a speech of only one speaker in a segment, not all data from the segment can contribute to the supervector equally. In a telephone conversation, crosstalk is frequent around the place of speaker change and also rapid changes of the speakers are common.

In Subsection 2.2, all statistics are accumulated into the supervector with the weight ω_m obtained only from the UBM. This weight ω_m in Equation (2) informs about the relevance of the acoustic data to "the universal speaker", in other words, how likely it is to be a part of a speech. This weight tells us nothing about the homogeneity of the speaker in the segment. Supervector accumulation, originally used in the speaker adaptation task, does not have to consider the homogeneity of the speaker in data.

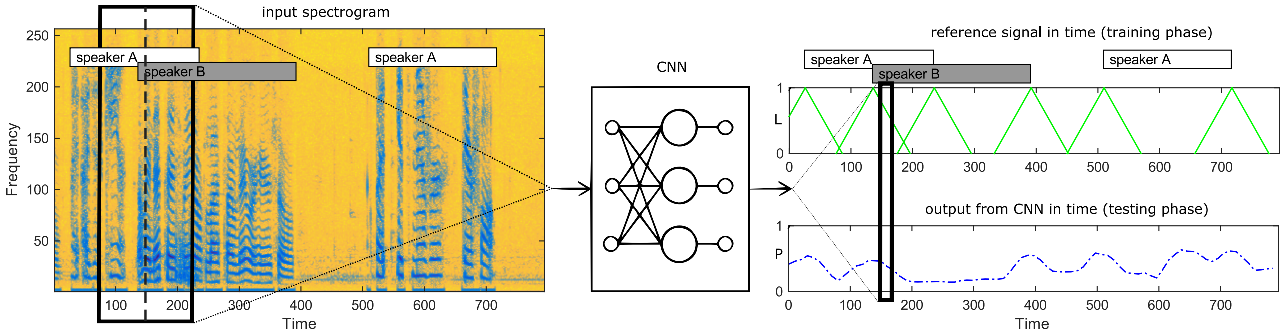

Figure 2: *The input speech as spectrogram is processed by the CNN into the output function P (a probability of change in time). The L-function (the reference speaker change) for the CNN training is depicted on top. Note: the output of CNN in time t is only a number.*

For this purpose, we are exploring the output of the CNN-based SCD as a probability of the speaker change in the signal. Although the audio signal is cut into segments according to the maxima peaks in the function P (the CNN output), the shape of the function can also indicate a suspicious part of the segment. 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 probability P_t. Thus, we use the value of $1 - P_t$ as a weighting factor of the signal in the accumulation process. The refinement of Equation (2) is represented by the formula

$$\gamma_m(\boldsymbol{o}_t) = \frac{(1 - P_t)\omega_m \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{\mu}_m, \boldsymbol{C}_m)}{\sum_{m=1}^{M} \omega_m \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{\mu}_m, \boldsymbol{C}_m)}. \qquad (6)$$

The equations (3) and (4) stay the same because they both depend on the refined $\gamma_m(\boldsymbol{o}_t)$ from the Equation (6). The amount of data for the statistics accumulation stay the same only the importance of each data is changed.

4. Discussion

The limitation of the segmentation step in the SD system is a minimal length of the segment from which the identity of the speaker can be extracted. In telephone conversations, the speaker change can occur arbitrarily often in time. In these conditions, the segments should be long enough to allow the extraction of speaker identifying information while limiting the risk of a speaker change being present within the segment. Still, only one speaker in the whole segment can not be always guaranteed. A high probability value of a speaker change from the CNN represents the instability of homogeneity of a speaker in the segment. This instability leads to the propagation of faulty features into the supervector accumulation process. Such faulty features usually occur on the boundaries of the segment, where a high risk of crosstalk is common or anywhere in the segment if some disturbance in the acoustic signal is present, see Figure 3. When using the CNN output for the refinement of the statistics accumulation we suppress the effect of these faulty features by weighting them down.

Nevertheless, there are still known limitations of our proposed approach. In rare situations, when the speaker change is missed by the SCD as seen in Figure 4, we will only penalize the features corresponding to boundaries and to the missed speaker change. Thus the segment will be described by features from two different speakers, resulting into inaccurate i-vector representation.

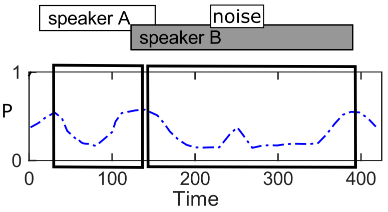

Figure 3: *Two speech segments with the probability of speaker change P, the first one with crosstalk on the end of the segment and the second one with noise disturbance in the middle of the segment.*

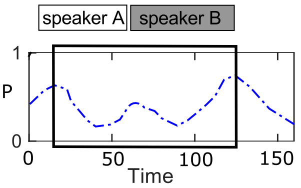

Figure 4: *Short speech segment with the probability of speaker change P containing two speakers. In this example, the SCD system fails and the P weight of statistics does not help to refine the accumulation process.*

The other SCD approaches (e.g. GLR used in [14]) have analogical output as the likelihood function of a speaker change. But for the purpose of weighting, the information from other SCD systems is inappropriate because usually the value of the change is not in the interval $\langle 0, 1 \rangle$ and the interval is changed for every conversation.

5. Experiments

The experiment was designed to investigate our proposed approach to refinement of the accumulation of statistics representing the speaker in the segment of conversation.

5.1. Corpus

The experiment was carried out on telephone conversations from the English part of CallHome corpus [34]. The original two channels have been mixed into one. Only two speaker conversations were selected so that the clustering can be limited to two clusters. This is 109 conversations in total each with about 10 min duration in a single telephone channel sampled at 8 kHz. For training of the CNN, only 35 conversations were used, the rest was used for testing the SD system.

5.2. System

The SD system presented in our papers [14, 35] uses the feature extraction based on Linear Frequency Cepstral Coefficients (LFCCs), Hamming window of length 25 ms with 10 ms shift of the window. There are 25 triangular filter banks which are spread linearly across the frequency spectrum, and 20 LFCCs are extracted. Delta coefficients were added leading to a 40-dimensional feature vector ($D_f = 40$). Instead of the voice activity detector, the reference annotation about missed speech was used.

For segmentation, CNN described in [14] was used. The input of the net is a spectrogram of speech of length 1.4 seconds and the shift is 0.1 seconds. The CNN consists of three convolutional layers with ReLU activation functions. There is a max-pooling layer after each convolutional layer. Batch normalization [36] is used for layer output normalization. There are two fully connected layers with sigmoid activation function at the end. In the first convolutional layer, there are filters with rectangular shapes that serve as feature extractors. The two intermediate convolutional layers learn a higher level representation of these features. The output layer consists of just one neuron with sigmoid activation function. Thus the output is limited between zero and one. It represents the probability of a speaker change in the middle of the observed spectrogram. For the training of the CNN, we use a Binary Cross Entropy loss function. It is optimized by Stochastic Gradient Descent with a batch size of 64. The learning rate is changed after a fixed number of iterations by a factor of 0.1. When the loss function is stabilized we use RMSProp algorithm for fine tuning of the network's weights.

For the purpose of training the i-vector we have used the following corpora: NIST SRE 2004, NIST SRE 2005, NIST SRE 2006 speaker recognition evaluations [37, 38, 39] and the Switchboard 1 Release 2 and Switchboard 2 Phase 3 [40, 41]. We model the UBM as a GMM with $M = 1024$ components. We have set the dimension of the i-vector to $D_w = 400$ and we have used the conversational dependent PCA to reduce the dimension further. We use eigenvectors with the ratio of their eigenvalue mass $p = 0.5$. We have used K-means clustering with cosine distance to obtain the speaker clusters.

5.3. Results

We use the Diarization Error Rate (DER) for the evaluation of our approach. It has been described and used by NIST in the RT evaluations [42]. We use the standard 250 ms tolerance around the reference boundaries. DER is a combination of several types of errors (missed speech, mislabeled non-speech, incorrect speaker cluster). We assume the information about the silence in all testing audios is available and correct. That means that our results represent only the error of incorrect speaker clusters. The results of the examined systems are shown in Table 1. For comparison, the result of segmentation using only constant length window is also shown. Using this approach a conversation is divided into short segments and the system then relies on the clustering and further resegmentation to refine the boundaries.

Table 1: *DER [%] of the SD systems with the i-vector speaker representation with constant length window segegmentation and SCD based on CNN (with and without refined statistics accumulation).*

system	DER [%]
Constant length window seg.	9.23
CNN-SCD without refinement	9.31
CNN-SCD with refinement	**7.84**

The difference in the results of the system using CNN-SCD without refinement and system using only the constant length window segmentation is small because of the resegmentation step, which repairs the inaccurate segmentation produced by the constant length window [14]. The effect of resegmentation is strong because there is sufficient amount of data available in each conversation for efficient training of GMM. However, our proposed approach to refined statistics accumulation using the output from the CNN-based SCD brings a more precise information to the speaker description. This improvement can be seen on the final DER of the system even after resegmentation step.

6. Conclusions

Most of the DNN based SD systems introduced in Section 1 use DNN to describe a speaker in a relatively short segment of conversation and then compare two representations of adjacent segments (e.g. so called d-vectors [12]) to decide if the speaker change occurred. On the contrary, our approach using the CNN-based SCD finds the possible speaker changes in spectogram and additionally uses the information for the refinement of accumulation process of statistics. These refined statistics represent the speaker information in the segment better than the classical approach to the statistics accumulation, so the computed i-vector is more precise and the final diarization error of the whole SD system is reduced. Our next goal is to train the CNN to represent the probability of the speaker homogeneity in the acoustics signal instead of the probability of the speaker change. Also, we want to replace the i-vector with a DNN-based vector and use the CNN probability of the speaker change as a prior when constructing this vector.

7. Acknowledgements

The work was supported by the Ministry of Education, Youth and Sports of the Czech Republic project No. LO1506. Access to computing and storage facilities (CESNET LM2015042) is greatly appreciated.

8. References

[1] X. A. Miro, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker Diarization: A Review of Recent Research," *Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356–370, 2012.

[2] M. Rouvier, G. Dupuy, P. Gay, E. Khoury, T. Merlin, and S. Meignier, "An Open-source State-of-the-art Toolbox for Broadcast News Diarization," in *Interspeech*, Lyon, 2013, p. 5.

[3] 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.

[4] S. H. Shum, N. Dehak, R. Dehak, and J. R. Glass, "Unsupervised Methods for Speaker Diarization: An Integrated and Iterative Approach," *Audio, Speech, and Language Processing*, vol. 21, no. 10, pp. 2015–2028, 2013.

[5] 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.

[6] C. Fredouille, S. Bozonnet, and N. Evans, "The LIA-EURECOM RT 09 Speaker Diarization System," in *NIST Rich Transcription Workshop (RT09)*, Melbourne, USA, 2009.

[7] O. Ben-Harush, O. Ben-Harush, I. Lapidot, and H. Guterman, "Initialization of Iterative-Based Speaker Diarization Systems for Telephone Conversations," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 414–425, 2012.

[8] A. G. Adami, S. S. Kajarekar, and H. Hermansky, "A New Speaker Change Detection Method for Two-Speaker Segmentation," in *ICASSP*, vol. 4, 2002, pp. 3908–3911.

[9] J. Ajmera, I. McCowan, and H. Bourlard, "Robust Speaker Change Detection," *Signal Processing Letters, IEEE*, vol. 11, pp. 649–651, 2004.

[10] B. Fergani, M. Davy, and A. Houacine, "Speaker Diarization Using One-Class Support Vector Machines," *Speech Communication*, vol. 50, no. 5, pp. 355–365, 2008.

[11] V. Gupta, "Speaker Change Point Detection Using Deep Neural Nets," in *ICASSP*, Brisbane, 2015, pp. 4420–4424.

[12] R. Wang, M. Gu, L. Li, M. Xu, and T. F. Zheng, "Speaker Segmentation Using Deep Speaker Vectors for Fast Speaker Change Scenarios," in *ICASSP*, New Orleans, 2017, pp. 5420–5424.

[13] S. Furui and D. Itoh, "Neural-Network-Based HMM Adaptation for Noisy Speech," in *ICASSP*, Salt Lake City, 2001, pp. 365–368.

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

[15] J. R. Hershey, Z. Chen, J. L. Roux, and S. Watanabe, "Deep Clustering: Discriminative Embeddings for Segmentation and Separation," in *ICASSP*, Shanghai, 2016, pp. 31–35.

[16] R. Milner and T. Hain, "DNN-Based Speaker Clustering for Speaker Diarisation," in *Interspeech*, vol. 08-12-Sept, San Francisco, 2016, pp. 2185–2189.

[17] G. Sell, D. Garcia-Romero, and A. Mccree, "Speaker Diarization with I-Vectors from DNN Senone Posteriors," in *Interspeech*, Dresden, 2015, pp. 3096–3099.

[18] S. H. Yells, A. Stolcke, and M. Slaney, "Artificial Neural Network Features for Speaker Diarization," in *Proc. IEEE Spoken Language Technology Workshop*. IEEE, 2014, pp. 402–406.

[19] N. Dawalatabad, S. Madikeri, C. C. Sekhar, and H. A. Murthy, "Two-Pass IB Based Speaker Diarization System Using Meeting-Specific ANN Based Features," in *Interspeech*, San Francisco, 2016, pp. 2199–2203.

[20] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker Diarization Using Deep Neural Network Embedings," in *ICASSP*, New Orleans, 2017, pp. 4930 – 4934.

[21] H. Bredin, "TristouNet: Triplet Loss for Speaker Turn Embedding," in *ICASSP*, New Orleans, 2017, pp. 5430–5434.

[22] M. Hrúz and M. Kunešová, "Convolutional Neural Network in the Task of Speaker Change Detection," in *Specom*. Budapest: Springer International Publishing, 2016, pp. 191–198.

[23] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1106–1114.

[25] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[26] 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.

[27] Z. Zajíc, L. Machlica, and L. Müller, "Robust Adaptation Techniques Dealing with Small Amount of Data," in *TSD 2012. Lecture Notes in Computer Science*, vol. 7499, Brno, 2012, pp. 418–487.

[28] ——, "Robust Statistic Estimates for Adaptation in the Task of Speech Recognition," in *TSD 2010. Lecture Notes in Computer Science*, vol. 6231. Brno: Springer, Berlin, Heidelberg, 2010, pp. 464–471.

[29] P. Kenny and P. Dumouchel, "Experiments in Speaker Verification Using Factor Analysis Likelihood Ratios," in *Odyssey - Speaker and Language Recognition Workshop*, Toledo, 2004, pp. 219–226.

[30] P. Kenny, "Joint Factor Analysis of Speaker and Session Variability: Theory and Algorithms," Tech. Rep., 2006.

[31] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of I-vector Length Normalization in Speaker Recognition Systems," in *Interspeech*, Florence, 2011, pp. 249–252.

[32] 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.

[33] L. Machlica and Z. Zajíc, "Factor Analysis and Nuisance Attribute Projection Revisited," in *Interspeech*, Portland, 2012, pp. 1570–1573.

[34] A. Canavan, D. Graff, and G. Zipperlen, "CALLHOME American English Speech, LDC97S42," in *LDC Catalog*. Philadelphia: Linguistic Data Consortium, 1997.

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

[36] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *Arxiv*, vol. abs/1502.0, 2015.

[37] A. Martin and M. Przybocki, "2004 NIST Speaker Recognition Evaluation, LDC2006S44," in *LDC Catalog*. Philadelphia: Linguistic Data Consortium, 2011.

[38] NIST Multimodal Information Group, "2005 NIST Speaker Recognition Evaluation Training Data, LDC2011S01," in *LDC Catalog*. Philadelphia: Linguistic Data Consortium, 2011.

[39] ——, "2006 NIST Speaker Recognition Evaluation Training Set, LDC2011S09," in *LDC Catalog*, 2011.

[40] D. Graff, D. Miller, and K. Walker, "Switchboard-2 Phase III Audio," in *LDC Catalog*. Philadelphia: Linguistic Data Consortium, 1999.

[41] D. Graff, K. Walker, and A. Canavan, "Switchboard-2 Phase II, LDC99S79," in *LDC Catalog*. Philadelphia: Linguistic Data Consortium, 2002.

[42] 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.