Interspeech 2018
2-6 September 2018, Hyderabad

# Joint Discriminative Embedding Learning, Speech Activity and Overlap Detection for the DIHARD Speaker Diarization Challenge

*Valter A. Miasato Filho*[1], *Diego A. Silva*[1], *Luis Gustavo D. Cuozzo*[1,2]

[1]CPqD, Campinas, Sao Paulo, Brazil
[2]Federal University of Technology, Curitiba, Parana, Brazil
{valterf,diegoa}@cpqd.com.br, lcuozzo@alunos.utfpr.edu.br

## Abstract

The DIHARD is a new, annual speaker diarization challenge focusing on "hard" domains, i.e. datasets in which current state-of-the-art systems are expected to perform poorly. We present our diarization system, which is a neural network jointly optimized for speaker embedding learning, speech activity and overlap detection. We present our network topology and the affinity matrix loss objective function responsible for learning the frame-wise speaker embeddings. The outputs of the network are then clustered with KMeans, and each frame classified with speech activity is assigned to one or two speakers, depending on the overlap detection. For the training data, we used two well-know meeting corpora - the AMI and the ICSI datasets, together with the provided samples from the DIHARD challenge. To further enhance our system, we present three data augmentation settings: the first is a naive concatenation of isolated speaker utterances from non-diarization datasets, which generates artificial diarization prompts. The second is a simple noise addition with sampled signal-to-noise ratios. The third is using noise suppression over the development data. All training setups are compared in terms of diarization error rate and mutual information in the evaluation set of the challenge.

**Index Terms**: speaker diarization, speaker clustering, speaker embeddings, speech activity detection, speech overlap detection

## 1. Introduction

Speaker diarization is the task of identifying speakers in an audio prompt and annotating the position and duration of each of their utterances, solving the "who spoke when" problem. Typically the problem assumes that no information about the identity or number of speakers is present in inference time. The diarization task has been solved independently in different domains, such as telephony, broadcast news and meetings [1]. Their differences induce domain-specific heuristics, especially when resources such as multiple microphones [2, 3] and video recordings are also available [4]. The DIHARD [5] challenge provides a single-channel multi-domain speaker diarization evaluation set, a scenario which supposedly is much more challenging for the current state-of-the-art.

A typical diarization pipeline is divided in subtasks, such as speech activity and overlap detection, speaker representation and clustering, and in some systems more tasks like speaker turn detection [6] gender identification [7], re-segmentation [8] and speech enhancement [9] are also employed for robustness. In our previous work in [10], we argued for a simpler diarization pipeline in inference time. We proposed a new form of speaker representations through frame-wise speaker embeddings which could be clustered with computationally efficient algorithms such as the *k-means*. The embeddings were derived from the work in speech separation in [11].

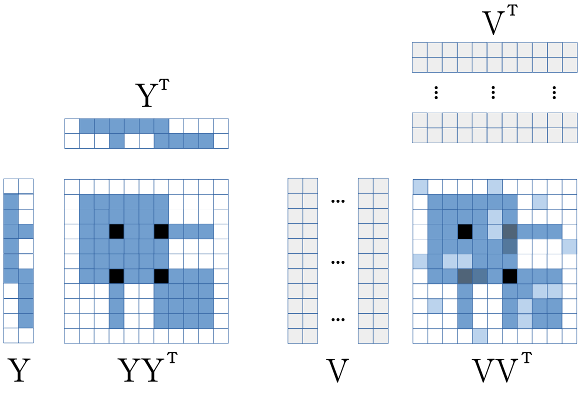

Figure 1: *Visual representation of the affinity matrices.*

This proposition was aligned with the current trend of generating speaker embeddings, either through speaker classification tasks and transfer learning [6] or through speaker verification tasks with similar discriminant loss functions [8, 12]. In this work, we contextualize our proposition, update its formulation and highlight its key differences in section 2.

We also propose time-convolving neural networks instead of the prior recurrent layers in [10], motivated by the independence of the context size (number of timesteps) between training and inference, and also by speed and memory consumption. Another important part of this work is the exploration of external datasets and data augmentation strategies due to the data-intensive nature of neural network training. The data and all propositions are presented in sections 3 and 4.

The remaining sections serve the following purposes: in section 5, we briefly reference our simplified diarization pipeline. In section 6 we show the outcome and analysis of our experiments and in section 7 we make our final remarks and present our expectations and future plans.

## 2. Previous work

### 2.1. Speaker embeddings

The cost function for modeling speaker embeddings is based on the described in [10], which is the same cost function for modeling time-frequency bins in speech separation described in [11]. Instead of focusing on the standalone definition of the affinity matrix loss function, in this work we will compare its formulation to the contrastive loss [13], which also inspires other loss functions in speaker embedding generation [14, 15, 12].

The embeddings have frame-wise resolution, i.e. are assigned individually to each window $t$ from a short-time Fourier

10.21437/Interspeech.2018-2304

transform. Assuming each vector embedding $V_t$ has a size of $K$, there is a corresponding reference $Y_t$ which is a one-hot representation of the speakers present in the current speech segment. The loss function is defined as:

$$C(Y,V) = ||VV^T - YY^T||_F^2, \tag{1}$$

in which $||M||_F^2$ is the squared Frobenius norm of a matrix $M$. This formulation is equivalent to a series of element-wise dot product differences which can be visualized in Figure 1, where squares are elements of n-dimensional vectors, whites represent 0, blues represent 1, blacks represent 2, and grays are undefined. The equation may be expanded to its individual values:

$$C(Y,V) = \sum_{i=1}^{T}\sum_{j=1}^{T}(v_i \bullet v_j - y_i \bullet y_j)^2, \tag{2}$$

in which each pair $i, j$ represents two aforementioned windows from a STFT, and all pairs are compared in an all-against-all fashion inside a pre-specified timestep of length $T$. For simplicity, we will first consider each $y$ having only one active speaker, with $y_i \bullet y_j$ being equal to 1 if both time frames represent the same speaker, and 0 otherwise. We may then expand each dot product difference to a local pairwise conditional cost:

$$C(Y,V)_{i,j} = \begin{cases} (v_i \bullet v_j)^2, & \text{if } i,j \text{ are from} \\ & \text{different speakers} \\ (1 - v_i \bullet v_j)^2 & \text{otherwise.} \end{cases} \tag{3}$$

If $v_i, v_j$ have unitary norm, this formulation is equivalent to the contrastive loss if the euclidean distance is switched by the cosine dissimilarity. The dot product is naturally bounded by the constrained norms of the input vectors, rendering cost function always non-negative. However, we still risk our same-class examples collapsing in a single point, so we propose a different margin $0 <= m < 1$ to be added to the local cost:

$$C(Y,V)_{i,j} = \max((v_i \bullet v_j - y_i \bullet y_j)^2 - m, 0). \tag{4}$$

The original contrastive margin sets a maximum radius for different-pairs, assuring the non-negativity of the loss function [13]. The proposed margin instead defines an angular region in which the embeddings are assumed to be in their optimal position (loss 0). The margin is a novelty when comparing to the previous work in [10]. The low-rank formulation of the affinity matrix in [11] in this case is not directly applicable due to the vector-wise maximum operator. For us, this was not a concerning issue, since we were able to use the high-rank formulation within our memory constraints.

The final formulation of the affinity matrix loss accounts for frames $i$ in which the number of present speakers is different from 1 (silence and speech overlap frames). In these cases, $|y_i| \neq 1$, and we adjust the norm of $v_i$ accordingly:

$$C(Y,V)_{i,j} = \max((|y_i|v_i \bullet |y_j|v_j - y_i \bullet y_j)^2 - m, 0). \tag{5}$$

With this new formulation, the cost is zero when either $i$ or $j$ represents a silence region, and thus we avoid learning embeddings for silence. Also, when two or more speakers are present, the embeddings are pulled to the angular average of the represented speakers.

Aside from formulation, there are other two key features from the embeddings learned via affinity matrix loss:

Table 1: *Development datasets.*

| Domain | Duration | Speech% | Ovp% | Spk# |
|---|---|---|---|---|
| AMI | 75:39:25 | 85.8 | 16.3 | 3 to 6 |
| ICSI | 71:41:12 | 85.6 | 15.2 | 3 to 10 |
| DIHARD | 18:56:50 | 76.1 | 6.3 | 1 to 10 |
| SEEDLINGS | 1:50:58 | 60.1 | 9.3 | 2 to 5 |
| SCOTUS | 2:04:46 | 84.0 | 1.6 | 5 to 10 |
| DCIEM | 2:29:58 | 68.5 | 2.0 | 2 |
| ADOS | 2:10:12 | 61.0 | 2.3 | 2 to 3 |
| YP | 2:03:25 | 78.5 | 1.0 | 3 to 5 |
| SLX | 2:00:26 | 72.4 | 5.7 | 2 to 6 |
| VAST | 1:50:20 | 85.7 | 11.8 | 1 to 9 |
| RT04S | 2:26:15 | 93.7 | 21.7 | 3 to 10 |
| LIBRIVOX | 2:00:30 | 79.4 | 0.0 | 1 |

- The embeddings have frame-wise resolution;
- The positive/negative examples are sampled from the same context, i.e. are speakers in the same conversation within a time frame.

The implications of those features, their potential strengths and pitfalls are discussed in section 6.

### 2.2. Joint optimization

The original motivation for joint optimization in [10] was the usage of traditional loss outputs for regularization. From then on, a couple of differences arose in comparison to prior work, which had significant impact in training stability:

- Angular margin for the affinity matrix loss;
- Unit norm constraint in the embedding layer;
- Switch from recurrent to time-convolutional layers;
- Layer-wise batch normalization.

With those, the most compelling reason to maintain the joint optimization was the time saved by training a single network instead of multiple ones. The comparison of multi-output networks to multiple single-output networks is out of scope of this work. The first two items were presented in 2.1, while the later two are explained in section 4.

## 3. Datasets

### 3.1. DIHARD development dataset

The DIHARD development dataset [16] [17] was distributed to registered participants for the development and training of the diarization systems. The DIHARD dataset has approximately 19 hours worth of 5-10 minute 16kHz, monaural prompts in 165 FLAC files, comprising a variety of domains shown in Table 1. The collection of all domains were split into training, validation, and test sets with the respective approximate ratios of 50%, 25% and 25%, ensuring that all domains were present under the three partitions.

### 3.2. Additional development datasets

We used the publicly available AMI [2] and ICSI [3] datasets as additions to the provided development data from the DIHARD challenge. Both datasets are composed of multi-party meeting recordings, from which we used the headset mixes as monaural data. The AMI corpus had poor quality in their original mix due to the noise in some channels being louder than speech in others, so we used the SoX tool [18] to apply dynamic range compression and amplitude normalization in the individual channels before mixing. We split those datasets in training, validation and test sets in roughly estimated proportions of 80%, 10% and 10%, respectively. All three partitions have disjoint sets of speakers.

For augmenting our training set, we also used the Voxceleb [19] set, which annotates celebrity speech in web videos. The augmentation scheme is described in section 4.4.

### 3.3. DIHARD evaluation data

The evaluation data consists of around 21 hours of data with the same characteristics of the development set, except by the addition of a new domain, consisting of recordings from conversations in restaurants. The same set was used in two different tracks for the challenge: diarization from gold speech segmentation (Track1) and diarization from scratch (Track2).

## 4. Training

All our models are trained with the same parameters and differ only by the data provided, either by the artificial prompts explained in section 4.3 or the data augmentation in 4.4.

### 4.1. Neural network topology

Our network is comprised of seven time-convolving layers. Each layer is described in Figure 2, with $w$ standing for the width of the convolution and $d$ for the dilation. All layers have $D = 512$ filters for convolution, and have *ReLUs* as non-linearities. Batch normalization [20] is applied in between layers for more stable training and faster convergence. To avoid fine-tuning gradient descent parameters, the Adam optimizer [21] was employed and gradients were clipped for having the maximum norm of 1.

The outputs of the network are all connected to the last layer with time-distributed weights. The embedding output is a fully connected layer with $K = 100$ activations constrained by the sigmoid non-linearity. The final vectors are then divided by their norm. The SAD and overlap outputs are both single sigmoids for binary classification.

### 4.2. Training examples

The input of our network is the log spectrum of the audio prompts in which speaker diarization is to be performed. We chose a window of $25ms$ with a shift of $30ms$ to perform the short-time Fourier transform. This configuration was inspired by [22] and was used for faster learning and inference. The block size in number of timesteps was $T = 1024$, which accounts for roughly $30s$ of context.

The vector $Y$ has 10 columns, which accounts for the maximum number of speakers in a single audio prompt seen in the training set. The one-hot activations are then determined by the order of appearance of different speakers in a single prompt. We activate the positions for all overlapping speakers in case of speech overlap. Two vectors $S$ and $O$ are also generated for speech activity and overlap detection. For all $y_i \in Y$, $1 < i < T$ in which at least 1 speaker is present, we define $s_i = 1$, $s_i \in S$, and 0 otherwise. In the same way for speech overlap, $o_i = 1$, $o_i \in O$ when at least 2 speakers are present, and 0 otherwise.

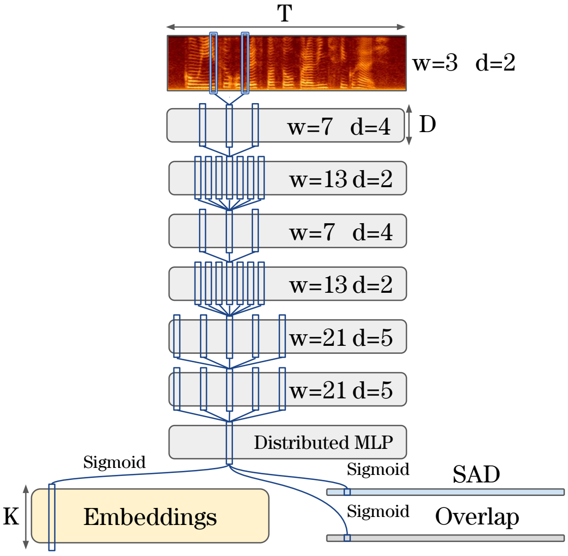

Figure 2: *Neural network topology.*

Vectors $S$ and $O$ are compared against the neural network output using the binary cross-entropy loss, while the $Y$ vector is compared to the embedding output $V$ via the affinity matrix loss, with $m = 0.2$ as the value of the angular margin.

For balancing speech activity and overlap data, we apply sample weights based on a running ratio of the amount of positive/negative examples. This is especially important in the speech overlap task, in which there are much more negative examples than positive ones.

We sample 512 batches of 64 examples from different files through 200 iterations, each taking an average of $3880s$ to complete. Intermediate models are saved each epoch and the one with the lowest validation DER is the final model.

### 4.3. Artificial prompt generation

For generating the artificial prompts with the VoxCeleb [19] dataset, we used a simple scheme of sampling at most 2 utterances from a range of 3 to 10 speakers from the dataset. The final number of examples is then $6 <= s <= 20$. From each utterance, a random sample of length $l <= T/s$ is extracted, and all samples are concatenated to form an input X of length $T$. The output $Y$ is generated with the speaker information from the dataset. Outputs $S$ and $O$ have their sample weights set to 0 to avoid learning speech activity and overlap from the artificial prompts. In training time, half of the batch is occupied by these sampled artificial prompts.

### 4.4. Data augmentation

We applied two data augmentation techniques for our datasets: noise addition in the AMI and ICSI datasets and noise suppression in the samples provided for the DIHARD challenge. The noise addition was performed with the FaNT tool [23], using external noise samples. We applied CHIME3 [24] samples over AMI, and QUT-NOISE-TIMIT [25] samples over ICSI, both with random signal-to-noise ratios between 5dB and 15dB. The noise supression was used with the corresponding module from the WebRTC project [26] in the DIHARD development set.

# 5. Diarization system

The diarization system which uses our trained neural network is the same as described in [10], which uses *k-means* for speaker clustering and a simple endpoint detection algorithm for generating the final speaker segments. There are two differences to the original formulation: first we make a full inference over the audio prompt instead of making a succession of predictions with a fixed value of timesteps $T$. The second difference is the length of the endpoint detection window for triggering start and end of speech. Setting the length of the window to a single frame minimizes the diarization error rate in all tested scenarios.

# 6. Results and analysis

We present our results based on the data augmentation strategies employed in training, identified as follows:

- Systems B have the noise addition/suppression data augmentation strategies, whereas systems A have not;
- Systems 2 have the VoxCeleb artificial prompts as examples, whereas systems 1 have not.

With these, we compare four systems (A1, B1, A2, B2) in each of the subtasks and in terms of the final diarization results.

## 6.1. Speech activity and overlap detection

In Table 2 we present the results for speech activity and overlap detection tasks. For speech activity we show the missed speech (MS) and total error (TER) rates. For speech overlap the used metrics are the precision (positive predictive value, PPV), recall (true positive rate, TPR) and their harmonic mean, the F1-score.

Table 2: *Speech activity and overlap detection.*

| **AMI/ICSI - Test** | | | | | |
|---|---|---|---|---|---|
| | **SAD** | | **Overlap** | | |
| **System** | **MS** | **TER** | **PPV** | **TPR** | **F1** |
| A1 | **0.073** | 0.136 | 0.456 | 0.362 | 0.447 |
| B1 | 0.077 | **0.132** | 0.512 | 0.410 | 0.455 |
| A2 | 0.082 | 0.134 | **0.559** | 0.414 | **0.476** |
| B2 | 0.086 | **0.132** | 0.483 | **0.422** | 0.451 |
| **DIHARD - Test** | | | | | |
| A1 | 0.110 | 0.142 | 0.485 | 0.048 | 0.088 |
| B1 | **0.093** | **0.127** | **0.621** | **0.099** | **0.171** |
| A2 | 0.164 | 0.228 | 0.399 | 0.055 | 0.096 |
| B2 | 0.139 | 0.191 | 0.509 | 0.073 | 0.128 |

There is little difference between systems for the AMI and ICSI tests, which was expected due to the large amount of in-domain training data. In contrast, both in speech activity and overlap the B1 system consistently outperforms all others in the DIHARD test, showing that in this case, the data augmentation was fruitful. The addition of the VoxCeleb data does not to show positive results in this case since SAD and overlap are ignored in the artificial prompts. However, the explicit worsening should be investigated further.

## 6.2. Diarization error rate and mutual information

In table 3 we show the results for diarization error rate and mutual information for our test set. We also append the results for a combined system submitted to the challenge. The combined system comprises SAD and overlap detection from the B1 system and the embeddings from the B2 system.

Table 3: *Diarization error rate and mutual information.*

| **AMI/ICSI - Test** | | | | |
|---|---|---|---|---|
| | **Track1** | | **Track2** | |
| **System** | **DER (%)** | **MI** | **DER (%)** | **MI** |
| A1 | **38.34** | 5.04 | **51.29** | 4.60 |
| B1 | 40.90 | 5.05 | 53.69 | 4.59 |
| A2 | 39.40 | **5.13** | 52.01 | **4.67** |
| B2 | 38.47 | 5.08 | 51.83 | **4.67** |
| **DIHARD - Test** | | | | |
| A1 | 40.61 | 4.64 | 45.74 | 4.46 |
| B1 | 32.97 | **4.82** | **39.50** | **4.59** |
| A2 | 33.01 | 4.76 | 50.01 | 4.46 |
| B2 | **29.29** | 4.79 | 45.62 | 4.54 |
| **DIHARD - Eval** | | | | |
| Best | 42.28 | 8.08 | 48.85 | 7.88 |

Again, differences between systems for the AMI/ICSI tests are not significant. In the DIHARD test set, we were able to see gains from using the VoxCeleb dataset in track 1, which discards the SAD information. In track 2, the worse SAD has a significant impact in the final DER. The mutual information is an important metric for speakers with low occurrence in the test set, but their low variation shows that all systems should be fairly similar in this regard.

The results therefore show that all data augmentation strategies were fruitful at least to some extent. However, since the artificial prompts underperformed in terms of SAD/Overlap, our best system was actually a combination of B1 and B2 systems.

The results in the evaluation set were not on par with the best submissions, but are satisfactory given the simplicity of our diarization system. We also believe that this system should be able to leverage larger amounts of data for making diarization systems more robust in a wider range of domains.

# 7. Conclusions and future work

We presented a diarization system for the DIHARD challenge, using time-convolving neural networks with outputs for speaker embedding generation, speech activity and overlap detection. The embeddings were generated using a modified version of the affinity matrix loss. The embeddings were clustered with *k-means* and the final segments were generated with a simple endpoint detection algorithm. For training, we used the development data from the challenge together with a large amount of external data. We also applied data augmentation strategies, and showed that they are fruitful for enhancing our system.

The results in the evaluation show that there is still room for improvement, especially when we account for the simplicity of our diarization pipeline. We are interested in verifying the quality of our embeddings in larger diarization pipelines and more complex speaker clustering techniques. Also, we plan to validate the ability of our system to generalize to multiple scenarios with even more data. Finally, we had to combine SAD/overlap and speaker embeddings from different systems to achieve best performance. This shows that we need to evaluate individual networks for each of the aforementioned tasks.

# 8. References

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

[2] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, “The ami meeting corpus: A pre-announcement,” in *International Workshop on Machine Learning for Multimodal Interaction*. Springer, 2005, pp. 28–39.

[3] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke *et al.*, “The icsi meeting corpus,” in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP’03). 2003 IEEE International Conference on*, vol. 1. IEEE, 2003, pp. I–I.

[4] G. Friedland, H. Hung, and C. Yeo, “Multi-modal speaker diarization of real-world meetings using compressed-domain video features,” in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009, pp. 4069–4072.

[5] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, “First dihard challenge evaluation plan,” https://zenodo.org/record/1199638, 2018.

[6] M. Rouvier, P.-M. Bousquet, and B. Favre, “Speaker diarization through speaker embeddings,” in *Signal Processing Conference (EUSIPCO), 2015 23rd European*. IEEE, 2015, pp. 2082–2086.

[7] 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,” Idiap, Tech. Rep., 2013.

[8] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, “Speaker diarization using deep neural network embeddings,” in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017.

[9] S. Araki, M. Fujimoto, K. Ishizuka, H. Sawada, and S. Makino, “Speaker indexing and speech enhancement in real meetings/conversations,” in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*. IEEE, 2008, pp. 93–96.

[10] V. A. Miasato Filho, D. A. Silva, and L. G. D. Cuozzo, “Multi-objective long-short term memory neural networks for speaker diarization in telephone interactions,” in *2017 Brazilian Conference on Intelligent Systems (BRACIS)*. IEEE, 2017, pp. 181–185.

[11] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, “Deep clustering: Discriminative embeddings for segmentation and separation,” in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 31–35.

[12] H. Bredin, “Tristounet: triplet loss for speaker turn embedding,” in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5430–5434.

[13] R. Hadsell, S. Chopra, and Y. LeCun, “Dimensionality reduction by learning an invariant mapping,” in *Computer vision and pattern recognition, 2006 IEEE computer society conference on*, vol. 2. IEEE, 2006, pp. 1735–1742.

[14] K. Chen and A. Salman, “Extracting speaker-specific information with a regularized siamese deep network,” in *Advances in Neural Information Processing Systems*, 2011, pp. 298–306.

[15] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, “Deep neural network-based speaker embeddings for end-to-end speaker verification,” in *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, 2016, pp. 165–170.

[16] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, “Dihard corpus,” Linguistic Data Consortium, 2016.

[17] E. Bergelson, “Bergelson seedlings homebank corpus,” doi.org/10.21415/T5PK6D, 2016.

[18] L. Norskog and C. Bagwell, “Sox-sound exchange,” http://sox.sourceforge.net/, 2013, version 14.

[19] A. Nagrani, J. S. Chung, and A. Zisserman, “Voxceleb: a large-scale speaker identification dataset,” *arXiv preprint arXiv:1706.08612*, 2017.

[20] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *International conference on machine learning*, 2015, pp. 448–456.

[21] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[22] D. Povey, V. Peddinti, D. Galvez, P. Ghahrmani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, “Purely sequence-trained neural networks for ASR based on lattice-free MMI,” *Submitted to Interspeech*, 2016.

[23] H. G. Hirsch, “Fant: filtering and noise adding tool,” *Niederrhein University of Applied Sciences, http://dnt.-kr. hsnr. de/download. html*, 2005.

[24] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, “The third ‘chime’speech separation and recognition challenge: Dataset, task and baselines,” in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*. IEEE, 2015, pp. 504–511.

[25] D. B. Dean, S. Sridharan, R. J. Vogt, and M. W. Mason, “The qut-noise-timit corpus for the evaluation of voice activity detection algorithms,” *Proceedings of Interspeech 2010*, 2010.

[26] A. B. Johnston and D. C. Burnett, *WebRTC: APIs and RTCWEB protocols of the HTML5 real-time web*. Digital Codex LLC, 2012.