INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Self-Attentive Similarity Measurement Strategies in Speaker Diarization

Qingjian Lin[1,3], *Yu Hou*[1], *Ming Li*[1,2]

[1]Data Science Research Center, Duke Kunshan University, Kunshan, China
[2]School of Computer Science, Wuhan University, Wuhan, China
[3]School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China
ming.li369@dukekunshan.edu.cn

Abstract

Speaker diarization can be described as the process of extracting sequential speaker embeddings from an audio stream and clustering them according to speaker identities. Nowadays, deep neural network based approaches like x-vector have been widely adopted for speaker embedding extraction. However, in the clustering back-end, probabilistic linear discriminant analysis (PLDA) is still the dominant algorithm for similarity measurement. PLDA works in a pair-wise and independent manner, which may ignore the positional correlation of adjacent speaker embeddings. To address this issue, our previous work proposed the long short-term memory (LSTM) based scoring model, followed by the spectral clustering algorithm. In this paper, we further propose two enhanced methods based on the self-attention mechanism, which no longer focuses on the local correlation but searches for similar speaker embeddings in the whole sequence. The first approach achieves state-of-the-art performance on the DIHARD II Eval Set (18.44% DER after resegmentation), while the second one operates with higher efficiency.

Index Terms: speaker diarization, similarity measurement, self-attention, spectral clustering, DIHARD II

1. Introduction

Speaker diarization can be considered as the process of partitioning multi-speaker speech into short segments and clustering them according to speaker identities. It solves the "*who spoke when*" problem [1, 2], which has a wide range of applications in real-life scenarios such as meetings, telephone calls and child care.

A typical diarization system usually consists of multiple modules, as demonstrated in Figure 1. First, voice activity detection (VAD) detects speech in audio streams and removes the non-speech portions [3, 4, 5, 6]. Second, speaker changepoint detection (SCD) [7, 8] or uniform segmentation [9] partitions speech into multiple speaker-homogeneous segments. Third, speaker embeddings like i-vector [10], x-vector [11] and Deep ResNet vectors [12, 13] are extracted from the segments. Then similarity measurement algorithms such as cosine distance and PLDA [14, 15] compute scores between any two speaker embeddings in the sequence, followed by the clustering algorithms like agglomerative hierarchical clustering (AHC) [9, 16], K-means [17] and spectral clustering [17, 18] to generate the diarization outputs.

In general, conversations raised by speakers are highly structured, and turn-taking behaviors are not randomly distributed over time. When the speech regions are uniformly split into short segments, speaker embeddings extracted from adjacent segments are of high correlation. However, PLDA measures the similarity between arbitrary two speaker embeddings in a pair-wise and independent manner, namely vector-to-vector scoring, which ignores the positional correlation of the measured segments. In [18], we proposed a novel vector-to-sequence scoring model based on LSTM, and computed the similarity scores between one speaker embedding and the whole embedding sequence. The manner helped capture structural information from both forward and backward directions. During the past DIHARD II competition, the new scoring model was put into use and proved to be effective in challenging scenarios [19]. In this paper, we enhance the similarity measurement process with the self-attention mechanism. The first approach is an extension of the vector-to-sequence scoring manner, which replaces the LSTM structure with Transformer encoders while remaining input features and supervised target labels unchanged. Noticing that the vector-to-sequence scoring process is time-consuming, we further propose the second method: a faster sequence-to-sequence scoring model.

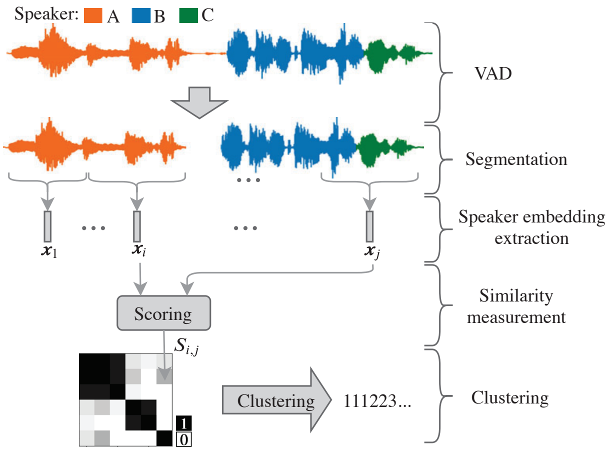

Figure 1: *A typical speaker diarization pipeline.*

The rest of this paper is organized as follows. Section 2 introduces the LSTM based scoring model and the spectral clustering algorithm. Section 3 describes our two proposed methods based on the self-attention mechanism. Experimental results and discussions are presented in Section 4, while conclusions are drawn in Section 5.

2. Related works

Suppose that $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n$ are a sequence of speaker embeddings extracted from an audio stream. In the similarity measurement stage, our goal is to construct the similarity matrix $\boldsymbol{S} \in \mathbb{R}^{n \times n}$, where $S_{i,j}$ is the similarity score between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

http://dx.doi.org/10.21437/Interspeech.2020-1908

2.1. PLDA

To construct the complete similarity matrix, the PLDA algorithm goes through all pairs of speaker embeddings $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. For each pair, it computes the corresponding score as follows:

$$S_{i,j} = f_{\text{PLDA}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i \boldsymbol{Q} \boldsymbol{x}_i + \boldsymbol{x}_i \boldsymbol{P} \boldsymbol{x}_j + \boldsymbol{x}_j \boldsymbol{Q} \boldsymbol{x}_j + const \quad (1)$$

Matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ are trainable parameters of PLDA, and *const* is a constant value. In this case, PLDA does not consider the information from neighborhoods of the two speaker embeddings.

2.2. LSTM based vector-to-sequence scoring

Speaker embeddings extracted from adjacent segments are more likely to be assigned to the same speaker, especially when the audio is segmented by uniform segmentation. To utilize this property, we proposed the LSTM based scoring model and computed similarity scores in a vector-to-sequence manner in [18]. Given the $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n$ sequence, $\boldsymbol{x}_i$ is selected and repeatedly concatenated with all of the speaker embeddings. Then the concatenated sequence is fed into the LSTM scoring model and generates corresponding similarity scores as the i-th row of the similarity matrix:

$$\boldsymbol{S}_i = [S_{i,1}, S_{i,2}, ...S_{i,n}] = f_{\text{LSTM}}\left(\begin{bmatrix}\boldsymbol{x}_i \\ \boldsymbol{x}_1\end{bmatrix}, \begin{bmatrix}\boldsymbol{x}_i \\ \boldsymbol{x}_2\end{bmatrix}, \cdots, \begin{bmatrix}\boldsymbol{x}_i \\ \boldsymbol{x}_n\end{bmatrix}\right). \quad (2)$$

Output $S_{i,j}$ denotes the similarity score of input concatenated vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, which is expected to be 1 for the same speaker and 0 for different speakers. To construct the complete similarity matrix $\boldsymbol{S}$, we run the model for n times with i ranging from 1 to n, and stack the outputs $\boldsymbol{S}_1, \boldsymbol{S}_2, ..., \boldsymbol{S}_n$ row by row vertically. For more details, please refer to [19].

2.3. Spectral clustering

Spectral clustering is a graph based clustering algorithm [20]. It regards $\boldsymbol{S}$ as an undirected graph and $S_{i,j}$ as the weight of the edge the between node i and j. By cutting off weak edges, the algorithm partitions the original graph into multiple subgraphs. Each subgraph represents a cluster. In this work, we employ spectral clustering as the back-end clustering method.

3. Proposed methods with self-attention

One limitation of the LSTM structure is that it focuses more on local information and may fail in long-term dependent tasks, although it has done much better than the conventional recurrent neural networks. To address this problem, Google proposed the Transformer structure with self-attention in language translation [21], which has been adopted in many different research fields recently. Since speakers who once spoke may appear again anywhere in the conversation, speaker diarization can also be categorized as a long-term dependent task. Therefore, we attempt to improve the model using the self-attention mechanism.

3.1. Attentive vector-to-sequence (Att-v2s) scoring

We keep inputs and supervised targets of the LSTM based method unchanged, and replace the neural network with the Att-v2s model. As depicted in Figure 2(a), the concatenated speaker embeddings are fully connected by the first linear layer and fed into two stacked encoder layers. Then feature mappings pass through the second linear layer with the Sigmoid function and generate similarity scores. The encoder layer is almost the same as the one in Speech-Transformer [22, 23], with the positional encoding layer removed. As shown in Figure 3, it mainly includes a multi-head self-attention module and a feed-forward module, both with layer normalization [24] and residual connection [25].

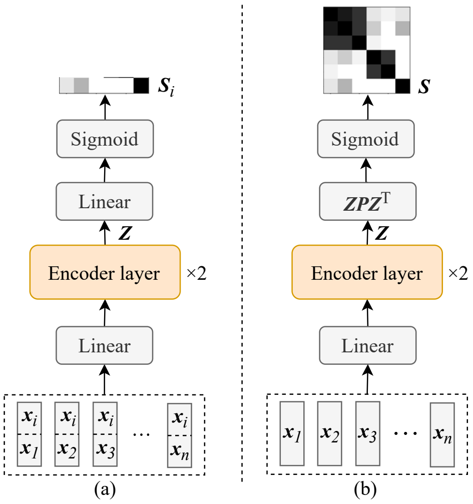

Figure 2: *(a) The attentive vector-to-sequence (Att-v2s) scoring model. (b) The attentive sequence-to-sequence (Att-s2s) scoring model.*

3.1.1. Multi-head self-attention module

The multi-head self-attention module consists of h parallelized self-attention heads. For the i-th head, feature mappings $\boldsymbol{E} \in \mathbb{R}^{n \times d}$ are converted into query matrix $\boldsymbol{Q}_i \in \mathbb{R}^{n \times d_q}$, key matrix $\boldsymbol{K}_i \in \mathbb{R}^{n \times d_k}$ and value matrix $\boldsymbol{V}_i \in \mathbb{R}^{n \times d_v}$ respectively by different linear layers. The Scaled-Dot Product Attention block copes with these three matrices as follows:

$$\text{head}_i = \text{Attention}(\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i) = \text{Softmax}(\frac{\boldsymbol{Q}_i \boldsymbol{K}_i^\top}{\sqrt{d_k}})\boldsymbol{V}_i. \quad (3)$$

Usually we set $d_q = d_k = d_v$ and the Softmax function is performed in a row-wise manner. Then results from different heads are concatenated over the last dimension and fully connected:

$$\text{MultiHead} = \text{Linear}(\text{Concat}(\text{head}_1, \text{head}_2, ..., \text{head}_h)). \quad (4)$$

3.1.2. Feed-forward module

The feed-forward module consists of two linear layers, with the ReLU function in between. The dimension of input and output keeps the same, and the inner-layer is high-dimensional.

3.2. Attentive sequence-to-sequence (Att-s2s) scoring

To construct the complete similarity matrix $\boldsymbol{S}$, we need to run the Att-v2s scoring model for n times. It is computationally expensive and increases the operation time of the whole diarization system. To deal with this issue, we propose the attentive sequence-to-sequence (Att-s2s) scoring model, which calculates the complete similarity matrix $\boldsymbol{S}$ in one shot.

As demonstrated in Figure 2(b), the overall structure of the Att-s2s scoring model is highly similar to the Att-v2s model. The main difference is that we replace the second linear layer with matrix production $\boldsymbol{Z}\boldsymbol{P}\boldsymbol{Z}^\top$, where matrix $\boldsymbol{P}$ is

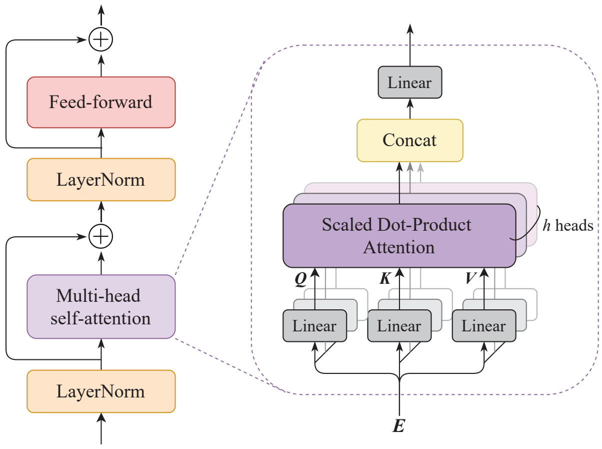

Figure 3: *Structure of the encoder layer.*

the trainable neural network parameters and initialized as identity matrix. Besides, the input is the original speaker embedding sequence instead of the concatenated one, and the output is directly the whole similarity matrix. Suppose that $\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, ..., \boldsymbol{z}_n]^\top \in \mathbb{R}^{n\times d}$, and the matrix production can be expanded as follows:

$$\boldsymbol{ZPZ}^\top = \begin{bmatrix} \boldsymbol{z}_1^\top \boldsymbol{P} \boldsymbol{z}_1 & \cdots & \boldsymbol{z}_1^\top \boldsymbol{P} \boldsymbol{z}_n \\ \vdots & \ddots & \vdots \\ \boldsymbol{z}_n^\top \boldsymbol{P} \boldsymbol{z}_1 & \cdots & \boldsymbol{z}_n^\top \boldsymbol{P} \boldsymbol{z}_n \end{bmatrix}. \qquad (5)$$

$\boldsymbol{z}_i^\top \boldsymbol{P} \boldsymbol{z}_j$ computes the similarity score between $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$, which is inspired and simplified from computation of PLDA in Eq. 1. We normalize the score to the range of (0, 1) by the Sigmoid function.

Without self-attentive encoder layers, the Att-s2s model degrades to the weighted inner-product. Thus we are interested in what the role of self-attention is in the process. As we know, speaker embeddings extracted from short utterances may be not representative enough for identifying a speaker due to the phonetic and other types of variabilities. To improve the robustness, we could choose longer segments for embedding extraction, or average the speaker embeddings from multiple short utterances of the same speaker. For diarization audios where multiple speakers are involved, we tend to adopt short segments to satisfy the speaker homogeneity. Therefore, the left solution is to average speaker embeddings. The self-attention mechanism makes it by searching for similar speaker embeddings over the whole sequence and summing them up with learnable weights. As depicted in Figure 4, speaker embeddings $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n$ are first transformed to $\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_n$ by the linear layer. At the t-th moment, $\boldsymbol{v}_t$ pays attention to the speaker embeddings that are of high correlation to itself, and assigns them with different weights $\boldsymbol{W}_t = [w_{t,1}, w_{t,2}, ..., w_{t,n}]$. The larger weight indicates higher similarity. By weighted sum of speaker embeddings $w_{t,1}\boldsymbol{v}_1 + w_{t,2}\boldsymbol{v}_2 + ... + w_{t,n}\boldsymbol{v}_n$, this mechanism generates a more robust speaker representation.

4. Experimental results and discussions

For simplicity, we employ the oracle VAD to remove non-speech regions from audio streams. Speech regions are segmented by uniform segmentation with the window of 1.5s length and 0.75s shift. Each segment is labeled with the most talkative speaker in the central 0.75s region.

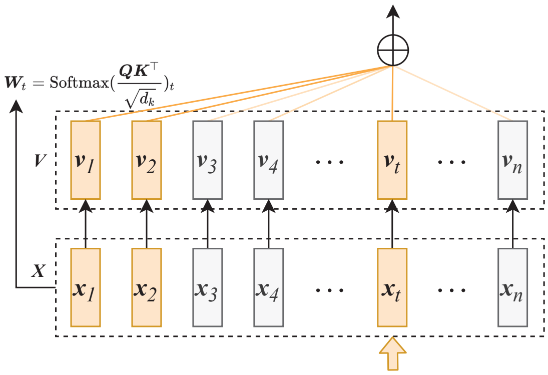
$\boldsymbol{W}_t = \text{Softmax}(\frac{\boldsymbol{QK}^\top}{\sqrt{d_k}})_t$
V
X
v_1
v_2
v_3
v_4
v_t
v_n
x_1
x_2
x_3
x_4
x_t
x_n

Figure 4: *Explanation of the self-attention mechanism in speaker diarization. Different colors denote different speakers.*

4.1. Speaker embedding extraction

Deep ResNet vector [13] is employed as the speaker embedding. Structure of Deep ResNet vector is similar to x-vector, but replaces the time delay neural network (TDNN) front-end with ResNet34. Specifically, it includes three main components: the ResNet34 front-end, the statistics pooling layer and the feed-forward layer. Detailed implementation can be referred in [13].

We take Voxceleb1&2 [26] corpora and their data augmentation for training, which includes 7323 speakers in total. 64-dimensional fbanks are extracted as the input features, with 25 ms length and 10 ms step. The dimension of Deep ResNet vector is 128.

Independent evaluation is carried out on the Voxceleb1_test Set and the equal error rate (EER) is reported. Besides full-length utterances (full-len), we also evaluate utterances limited to 1.5 seconds long (1.5s * 1). Moreover, we split the long utterance into multiple 1.5s segments, extract corresponding speaker embeddings, and then average them for the evaluation scenario (1.5s * N). Results are shown in Table 1. As expected, Deep ResNet vector in the full-length test condition achieves a low EER of 1.51%, and the performance degrades rapidly to 6.74% EER with the test duration limited to 1.5s. When we average multiple 1.5s speaker embeddings for each full-length utterance, the EER recovers to 1.98%.

Table 1: *Evaluation of Deep ResNet vector on Voxceleb1_test.*

	full-length	**1.5s * 1**	**1.5s * N**
EER(%)	1.51	6.74	1.98

4.2. Datasets

Public meeting corpora AMI [29] and ICSI [30] are employed for training the scoring models, about 170 hours in total. Audios are recorded in 16k sample rate, and the average duration is around 40 minutes. In the data preparation stage, thousands of 1.5s speaker embeddings are extracted from each audio. Then during the training stage, we randomly truncate 100 to 400 speaker embeddings from the selected audio.

Table 2: *Evaluation on DIHARD II corpus. Results are reported with and without domain adaptation by the Dev Set.*

Model	**+VB**	**Dev**		**Eval**		**Eval + adaptation**		**Time cost (Eval)**
		DER(%)	JER(%)	DER(%)	JER(%)	DER(%)	JER(%)	
LSTM	×	19.65	49.60	20.57	50.25	19.72	46.49	67 min
	√	19.48	49.21	19.98	49.42	19.26	45.91	-
Att-v2s	×	**19.07**	**47.43**	**20.15**	**47.84**	**18.98**	43.20	148 min
	√	**18.76**	**46.77**	**19.46**	**47.01**	**18.44**	42.52	-
Att-s2s	×	19.39	48.42	21.46	48.71	21.45	**43.19**	24 s
	√	19.16	47.99	20.78	47.92	20.12	**41.73**	-
PLDA	×	23.48	57.17	-	-	23.73	56.84	51 s
DIHARD II winner system [27]						18.42	44.58	
DIHARD II official baseline [28]						25.99	59.51	

Models are evaluated on the DIHARD II corpus [28], including the Dev Set and the Eval Set. Audios are sampled from 11 different domains with 16k sample rate, the number of speakers in each recording widely varies from 1 to 10. For vector-to-sequence scoring models, similarity matrices larger than the size of 400×400 are partitioned into submatrices and constructed accordingly.

4.3. Evaluation Metrics

We take the diarization error rate (DER) as the main metric, which consists of miss detection, false alarm and speaker error. There is no collar tolerance around speech turns, and miss detection of speakers in overlapped speech accounts for the error.

Another metric, namely Jaccard error rate (JER), is newly developed by the DIHARD competition [28]. It computes the sum of false alarm and miss detection for each individual in the audio, and then average the errors. Speakers with different duration contribute equally to the new metric.

4.4. Model configuration

In the Att-v2s model, the first linear layer is 256-dimensional. The encoder layer contains 2 heads with 128 attention units for each head, and the dimension of the feed-forward layer is 1024. The second linear is 1-dimensional, connected with the Sigmoid function. In the Att-s2s model, the same configuration is employed, except that we replace the second linear layer by matrix production $\boldsymbol{ZPZ}^{\top}$. Matrix $\boldsymbol{P}$ is the size of 256×256.

The binary cross entropy (BCE) loss function computes the loss between the similarity matrix and the ground truth adjacent matrix with binary values. The stochastic gradient descent (SGD) optimizer is employed, with the learning rate initialized as 0.01 and decreasing twice to 0.0001. The training process terminates after 100 epochs and we carry out evaluation on both Dev and Eval Sets (Dev/Eval in Table 2). Moreover, we fix the learning rate as 0.0001 and take the Dev Set to finetune the model for 30 more epochs. Then the adapted model is evaluated on the Eval Set (Eval + adaptation in Table 2).

4.5. Results

The spectral clustering algorithm is employed on top of similarity matrices to generate diarization outputs. Besides, we consider Variational Bayes (VB) resegmentation for enhancement of system performance. Results are reported in Table 2. All three neural network based models show significant improvement on both Dev and Eval Sets in comparison with PLDA. The best-performing single system, Att-v2s, achieves a DER of 18.44% on the Eval Set after domain adaptation and VB resegmentation, almost the same as that of the DIHARD II winner system. It is worth noting that the winner system includes an additional overlap detection module, which slightly reduces the error rate.

In most of the cases, both LSTM and Att-v2s models outperform the Att-s2s model. It is reasonable because the Att-s2s model constructs the whole similarity matrix in one shot, and the information it has to express is far more complicated than that of two vector-to-sequence models. As a compensation, the Att-s2s model runs in higher efficiency. It takes only 24 seconds to score the Eval Set on a single core of Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz, while the rest two models cost more than one hour.

Another interesting phenomenon is that the Att-s2s model gains the lowest JER on the Eval Set after domain adaptation, but meanwhile its DER is highest among the three models. Since speakers with different duration in the audio contribute equally to JER, even miss detection of the least talkative speaker raises a high error. We believe that JER is not as stable as DER, and thus report DER as the main metric.

5. Conclusions

In this paper, we review the LSTM based scoring model and propose two new methods with the self-attention mechanism. The first approach Att-v2s works in a vector-to-sequence manner, and achieves 18.44% DER on the DIHARD II Eval Set after domain adaptation and resegmentation. Besides, our second approach Att-s2s works in a sequence-to-sequence manner and operates with higher efficiency and comparable performance.

6. Acknowledgements

This research is funded in part by the National Natural Science Foundation of China (61773413), Key Research and Development Program of Jiangsu Province (BE2019054), Six talent peaks project in Jiangsu Province (JY-074), Science and Technology Program of Guangzhou, China (202007030011, 201903010040).

7. References

[1] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1557–1565, 2006.

[2] 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.

[3] C. Wooters, J. Fung, B. Peskin, and X. Anguera, "Towards robust speaker segmentation: The icsi-sri fall 2004 diarization system," in *Fall 2004 Rich Transcription Workshop*, 2004.

[4] R. Zazo Candil, T. N. Sainath, G. Simko, and C. Parada, "Feature learning with raw-waveform cldnns for voice activity detection," in *Interspeech*, 2016, pp. 3668–3672.

[5] 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 *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 483–487.

[6] S.-Y. Chang, B. Li, G. Simko, T. N. Sainath, A. Tripathi, A. van den Oord, and O. Vinyals, "Temporal modeling using dilated convolution and gating for voice-activity-detection," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 5549–5553.

[7] M. Hrúz and Z. Zajíc, "Convolutional neural network for speaker change detection in telephone speaker diarization system," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 4945–4949.

[8] R. Yin, H. Bredin, and C. Barras, "Speaker change detection in broadcast tv using bidirectional long short-term memory networks," in *Interspeech*, 2017, pp. 3827–3831.

[9] G. Sell and D. Garcia-Romero, "Speaker diarization with plda i-vector scoring and unsupervised calibration," in *2014 IEEE Spoken Language Technology Workshop*, 2014, pp. 413–417.

[10] S. Shum, N. Dehak, R. Dehak, and J. R. Glass, "Unsupervised speaker adaptation based on the cosine similarity for text-independent speaker verification." in *Odyssey 2010 The Speaker and Language Recognition Workshop*, 2010.

[11] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 5329–5333.

[12] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," in *Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018, pp. 74–81.

[13] W. Cai, J. Chen, J. Zhang, and M. Li, "On-the-fly data loader and utterance-level aggregation for speaker and language recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1038–1051, 2020.

[14] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *2007 IEEE 11th International Conference on Computer Vision*, 2007, pp. 1–8.

[15] P. Kenny, T. Stafylakis, P. Ouellet, M. J. Alam, and P. Dumouchel, "Plda for speaker verification with utterances of arbitrary duration," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7649–7653.

[16] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villalba, M. Maciejewski, V. Manohar, N. Dehak, D. Povey, S. Watanabe *et al.*, "Diarization is hard: Some experiences and lessons learned for the jhu team in the inaugural dihard challenge." in *Interspeech*, 2018, pp. 2808–2812.

[17] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. L. Moreno, "Speaker diarization with lstm," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 5239–5243.

[18] Q. Lin, R. Yin, M. Li, H. Bredin, and C. Barras, "Lstm based similarity measurement with spectral clustering for speaker diarization," in *Interspeech*, 2019, pp. 366–370.

[19] Q. Lin, W. Cai, L. Yang, J. Wang, J. Zhang, and M. Li, "Dihard ii is still hard: Experimental results and discussions from the dku-lenovo team," in *Odyssey 2020 The Speaker and Language Recognition Workshop*, 2020.

[20] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Neural Information Processing Systems*, 2017, pp. 5998–6008.

[22] L. Dong, S. Xu, and B. Xu, "Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 5884–5888.

[23] Y. Fujita, N. Kanda, S. Horiguchi, Y. Xue, K. Nagamatsu, and S. Watanabe, "End-to-end neural speaker diarization with self-attention," in *IEEE Automatic Speech Recognition and Understanding Workshop*, 2019, pp. 296–303.

[24] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[26] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," in *Interspeech*, 2017, pp. 2616–2620.

[27] F. Landini, S. Wang, M. Diez, L. Burget, P. Matejka, K. Žmolíková, L. Mosner, A. Silnova, O. Plchot, O. Novotny, H. Zeinali, and J. Rohdin, "But system for the second dihard speech diarization challenge," 05 2020, pp. 6529–6533.

[28] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, "The second dihard diarization challenge: Dataset, task, and baselines," in *Interspeech*, 2019, pp. 978–982.

[29] 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.

[30] 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 *2003 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2003.