DIVE: END-TO-END SPEECH DIARIZATION VIA ITERATIVE SPEAKER EMBEDDING

Neil Zeghidour, Olivier Teboul and David Grangier

Google Research, Brain Team

ABSTRACT

We introduce DIVE, an end-to-end speaker diarization system. DIVE presents the diarization task as an iterative process: it repeatedly builds a representation for each speaker before predicting their voice activity conditioned on the extracted representations. This strategy intrinsically resolves the speaker ordering ambiguity without requiring the classical permutation invariant training loss. In contrast with prior work, our model does not rely on pretrained speaker representations and jointly optimizes all parameters of the system with a multi-speaker voice activity loss. DIVE does not require the training speaker identities and allows efficient window-based training. Importantly, our loss explicitly excludes unreliable speaker turn boundaries from training, which is adapted to the standard collar-based Diarization Error Rate (DER) evaluation. Overall, these contributions yield a system redefining the state-of-the-art on the CALLHOME benchmark, with 6.7% DER compared to 7.8% for the best alternative.

Index Terms— diarization, speech, end-to-end learning.

1. INTRODUCTION

Speaker diarization is the task of annotating speaker turns in a conversation [1, 2, 3]. It is both a crucial step for downstream tasks such as automatic transcription of conversational speech, as well as a challenge as it requires handling long-term dependencies. Traditional systems typically split the problem in three sub-problems. First, a model is trained to extract short-term speaker embeddings. Such embeddings can be i-vectors derived from a Gaussian Mixture Model [4, 5, 6, 7, 8, 9], or embeddings produced by a neural network [10, 11, 12, 13, 14]. Then, given a sequence to be diarized, a pre-trained speech activity detection algorithm [15, 16] extracts active timesteps from the sequence and removes silences. Eventually, a clustering algorithm runs on top of these embeddings to assign each timestep to a speaker. Such composite systems have two main limitations. First, the speaker representations are not optimized for diarization, and may not extract relevant features for disambiguating speakers in e.g. presence of overlap. Moreover, most clustering algorithms being unsupervised, they cannot benefit from the fine-grained annotations of speaker turns in diarization datasets.

This motivated recent end-to-end diarization systems [17, 18]. In particular, [17, 19] propose to cast the diarization task as a multi-label classification problem. When trained to predict whether each speaker is active at each timestep, a single model jointly performs speech activity detection (silence vs speech), speaker modelling and clustering. This framework has been used to train various architectures including LSTMs [17] and self-attention [20] models [21]. Since diarization is a permutation-invariant problem (any permutation of the predicted speakers is valid), these models use Permutation-Invariant Training (PIT) [22, 19, 23] to avoid penalizing the model for choosing a particular speaker ordering. [24] has shown that PIT suffers from inconsistent assignments when applied to long sequences, and that it is preferable to explicitly learn long-term speaker representations. Moreover, fine-grained annotations can be unreliable around speaker turn boundaries. Hence, it is standard to remove the neighborhood of boundaries from evaluation [12]. As a consequence, inconsistent supervision around boundaries during training can adversely affect the final accuracy of the system.

In this work we introduce DIVE (Diarization by Iterative Voice Embedding), an end-to-end neural diarization system. DIVE combines three modules which are trained jointly: projection of the waveform to an embedding space, iterative selection of long-term speaker representations, and per-speaker per-timestep voice activity detection. The iterative speaker selection process addresses the problem of speaker order ambiguity and removes the need for training with PIT, similarly to attractor-based approaches [25, 26]. Moreover, we introduce collar-aware training, a modification to the standard multi-label classification loss which ignores errors in a defined radius around speaker turn boundaries to match the evaluation setting. DIVE obtains a state-of-the-art Diarization Error Rate (DER) of 6.7% on CALLHOME[27]. We also perform ablation studies that demonstrate the benefits of collar-aware training, and analyze the patterns of errors of our system.

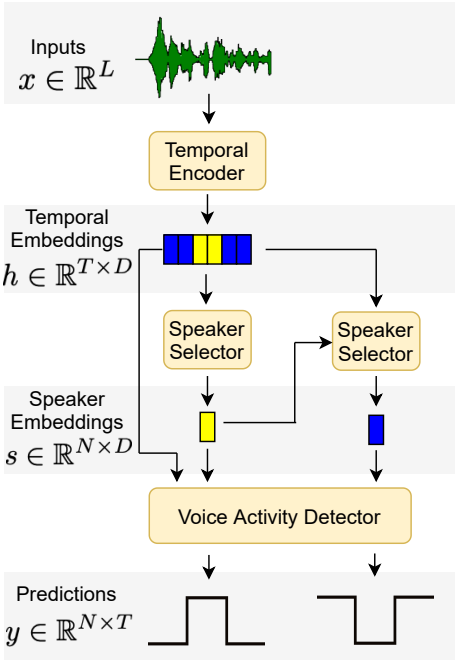

Fig. 1. DIVE for 2 speaker diarization. The temporal encoder first extracts a local speaker representation. The speaker selector iteratively selects the representation of a novel speaker when a single speaker is active. The voice activity module predicts speaker activity conditioned on the input signal and the selected representation.

2. METHOD

2.1. Setting and notations

We consider a single channel recording $x \in \mathbb{R}^L$ of N, partially overlapping, speakers, with L the length of the sequence. The goal of speaker diarization is to produce per-speaker voice activity masks $y_i \in \{0,1\}^T$ for $i = 1, \ldots, N$, with $y_{i,t} = 1$ meaning that speaker i is active at time t, and conversely. Typically, $T < L$ as the model does not produce voice activity masks at the sampling rate of the audio but rather at a lower sampling rate, e.g. every millisecond. DIVE cascades three components. First, a *temporal encoder* projects the input waveform to a downsampled embedding space. Then, the *speaker selector* identifies one embedding that characterizes well each speaker, in an iterative fashion. Eventually, the *voice activity detector* consumes the embeddings produced by the temporal encoder as well as the selected speaker embeddings and produces a binary voice activity mask for each speaker. We train these three modules jointly. In the following, we describe each component.

2.2. Temporal encoder

The *temporal encoder* projects the input waveform x to an embedding space, while performing downsampling. Precisely, the temporal encoder h produces T latent vectors of dimension D, i.e. $h(x) \in \mathbb{R}^{T \times D}$. We refer to these vectors as *temporal embeddings*. We use a temporal encoder similar to that of Wavesplit [24], which cascades residual blocks of dilated 1D-convolutions, with Parametric ReLU (PReLU) activations [28] and Layer Normalization [29]. Unlike the ReLU activation [30] which zeroes out negative values, the PReLU learns a per-channel slope parameter a:

$$\text{PReLU}(x) \quad = \quad \begin{cases} x & \text{if } x >= 0 \\ ax & \text{if } x < 0 \end{cases}$$

Unlike Wavesplit [24], where the temporal encoder maintains the original sampling rate of the signal, ours performs downsampling by introducing 1D average pooling layers between residual blocks. As the length of the audio sequence L varies between examples, training on batches requires either truncating or padding sequences to a standard length. Given a batch of sequences, a typical training scheme is to randomly sample a fixed-length window from each sequence and to batch the resulting segments [31, 24]. As such windows are typically short (a few seconds), they are likely to only contain one to two speaker turns. This is not appropriate for training a diarization system that needs to model transitions between speaker turns and maintain long term consistency in speaker assignments. To address this issue, we instead sample W fixed-length windows per sequence, pass them through the temporal encoder, and then concatenate them along the temporal axis. This allows for more diversity and more speaker turns inside a single training example. Section 3.4 assesses the advantage of multi-window training.

2.3. Iterative speaker selector

The role of the iterative speaker selector is to extract one speaker embedding vector $s_i \in \mathbb{R}^D$ for each active speaker in the signal. These vectors correspond to temporal embeddings extracted at time steps where a single speaker is active. To select these vectors, we iteratively compute the probability that a non-selected speaker is active. To do so, and at each time step, we define labels e_t^y of a 4-way classification problem among the following classes: a single novel speaker is active, a single already selected speaker is active, overlapped speech, or silence. More precisely, we extract these labels from voice

Algorithm 1 Iterative speaker selector

Inputs The number of speakers N, voice activity labels $y \in \{0,1\}^{N\times T}$, temporal embeddings $h \in \mathbb{R}^{T\times D}$, multilayered perceptrons $G_s : \mathbb{R}^D \to \mathbb{R}^{D\times 4}$ and $g_h : \mathbb{R}^D \to \mathbb{R}^D$, training $\in \{\text{True, False}\}$.

Outputs Speakers embeddings $s = (s_0, ...s_{N-1})$, speaker selection loss $\mathcal{L}_{\text{selector}}$.

$\mu_0 \leftarrow \mathbf{0}_D$ ▷ Average speaker embedding.
$s \leftarrow \emptyset$ ▷ Embeddings of the selected speakers so far.
$S \leftarrow \emptyset$ ▷ Indices of the selected speakers so far.
$\mathcal{L}_{\text{selector}} \leftarrow 0$ ▷ Loss.
for $i = 0$ to $N-1$ **do**
  $\mathrm{P}(e_t|h_t, s_0^{i-1}) = \mathrm{softmax}(G_s(\mu_i)^\top g_h(h_t))$
  if training **then**
    $e_t^y = \text{selector\_label}(y_{:,t}, S)$ ▷ Recompute speaker selector labels.
    $t_i^\star \sim \mathcal{U}\{t|e_t^y = \text{new\_speaker}\}$ ▷ Sample a frame with a new speaker.
    $S = S \cup \{j|y_{j,t_i^\star} = 1\}$ ▷ Add new ID to the set of selected speakers.
    $\mathcal{L}_{\text{selector}} \mathrel{-}= \frac{1}{TN}\sum_{t=1}^{T} \log \mathrm{P}(e_t = e_t^y|h_t, s_0^{i-1})$
  else
    $t_i^\star = \arg\max_t P(e_t = \text{new\_speaker}|h_t, s_0^{i-1})$ ▷ Select frame with highest confidence in a new speaker.
  $s = s \cup \{h_{t_i^\star}\}$
  $\mu_{i+1} = \frac{1}{i+1}\sum_j s_j$
return s, $\mathcal{L}_{\text{selector}}$

activity annotations $y_{i,t}$ as follows:

$$e_t^y = \text{selector\_label}(y_{:,t}, S) \quad (1)$$
$$= \begin{cases} \text{silence} & \text{if } \sum_j y_{j,t} = 0 \\ \text{new\_speaker} & \text{if } \sum_{j\notin S} y_{j,t} = 1 \text{ and } \sum_{j\in S} y_{j,t} = 0 \\ \text{overlap} & \text{if } \sum_j y_{j,t} > 1 \\ \text{selected} & \text{if } \sum_{j\notin S} y_{j,t} = 0 \text{ and } \sum_{j\in S} y_{j,t} = 1 \end{cases}$$

where S is the set of already selected speakers, initialized with $S = \emptyset$. At each iteration i, the model computes

$$P(e_t = \text{new\_speaker}|h_t, s_0^{i-1}) \quad (2)$$

given the temporal embeddings $h \in \mathbb{R}^{T\times D}$ and the embeddings of the already extracted speakers $s_0^{i-1} = (s_0, \dots s_{i-1})$. We then extract the next speaker embedding $s_i = h_{t_i^\star}$ where this computed probability is maximized,

$$t_i^\star = \arg\max_t P(e_t = \text{new\_speaker}|h_t, s_0^{i-1}). \quad (3)$$

We perform this iterative process for a fixed number of steps when the number of speakers is known (as in our experiments). Otherwise, the process is run until the confidence in a new speaker drops below a predefined threshold.

In practice, we compute the probability in Eq. (2) with a neural network classifier. The classifier combines two multilayered perceptrons (MLPs), i.e.

$$\mathrm{P}(e_t|h_t, s_0^{i-1}) = \mathrm{softmax}(G_s(\mu_i)^\top g_h(h_t)) \quad (4)$$

where μ_i denotes the average of $s_0, \dots s_{i-1}$, with the convention $\mu_0 = 0$. The first MLP G_s maps an averaged speaker embedding $\mu_i \in \mathbb{R}^D$ into a vector of $\mathbb{R}^{D\times 4}$ which is viewed as a D-by-4 matrix, while the second MLP maps a temporal embedding of $\mathbb{R}^D$ into a vector of $\mathbb{R}^D$.

During training, the parameters of the selector are trained by adding an auxiliary loss corresponding to the cross entropy on the 4-class problem of Eq. (4),

$$\mathcal{L}_{\text{selector}}(h, e^y) = -\frac{1}{TN}\sum_{t=1}^{T}\sum_{i=1}^{N} \log \mathrm{P}(e_t = e_{i,t}^y|h_t, s_0^{i-1})$$

where the ground truth $e_{i,t}^y$ is recomputed at each iteration using Equation 1. At training time, we also replace the selected time step $t_i^\star$ by a uniform sample among time steps where a novel speaker is active, which increases the robustness of the voice activity detector to variability in the speaker vectors. Finally, one should note that the iterative selection steps determine an order over the outputted speaker representations. At training time, we track the inferred order such that the correspondence between voice activity labels and speaker vectors is maintained. Algorithm 1 describes the entire process of iterative speaker selection during training and inference.

Our iterative speaker selection procedure has links with prior work in source separation. [32, 33] propose an iterative process to infer the list of speaker identities present in a recording for separation. These approaches require speaker identity labels for training. [34, 25] proposes to infer a sequence of latent speaker representation optimizing directly

separation performance. This idea has also been applied to diarization [26]. While attractive, these approaches make windowed training and generalization to longer test sequence difficult. In contrast, our work does not need training speaker identities and our method allows efficient window based training to generalize to long test sequences.

2.4. Voice activity detector

After selecting speaker embeddings, the last module of DIVE predicts the voice activity of each speaker $y_i \in \{0,1\}^T$ for $i = 1, \ldots, N$. The voice activity detector contains two parallel fully-connected neural networks f_h and f_s with PReLU [28] activations and Layer Normalization [29], except for the last layer which is a linear projection. To produce the voice activity $y_{i,t}$ of speaker i at timestep t, f_h and f_s project the current temporal embedding $h_t \in \mathbb{R}^D$ and the speaker vectors $[s_i; \overline{s}] \in \mathbb{R}^{2D}$ respectively:

$$\hat{y}_{i,t} = f_h(h_t)^\intercal f_s([s_i; \overline{s}]). \tag{5}$$

Here, $[s_i; \overline{s}]$ is the concatenation along the channel axis of s_i, the speaker vector of speaker i, and $\overline{s} = \frac{1}{N}\sum_{j=0}^{N-1} s_j$ the mean of all speaker vectors. Intuitively, this means that when predicting the voice activity of a speaker at a given time, we use three pieces of information: the temporal embedding that represents the current speech content, a speaker embedding that represents the identity of the speaker of interest, and another embedding that represents all speakers. The latter allows the classifier to exploit contrasts between the current speaker of interest and other speakers in the sequence.

During training, we cast the problem of per-speaker, per-timestep voice activity detection as independent binary classification tasks and backpropagate the following loss:

$$\mathcal{L}_{\text{vad}}(\hat{y}, y) = -\frac{1}{TN}\sum_{t=1}^{T}\sum_{i=1}^{N}\log(\sigma(\hat{y}_{i,t}(2y_{i,t}-1))). \tag{6}$$

2.5. Collar-aware training

When evaluating a diarization system in terms of DER, it is common to apply a *collar*, which is a tolerance around speaker boundaries such that the metric does not penalize the model for small annotation errors. A typical value for such a tolerance is 250ms on each side of a speaker turn boundary (500ms in total). Since we evaluate the model in these conditions, it would be beneficial to train it in a similar fashion i.e. to ignore errors within the collar tolerance. Thus, and as an additional contribution to the DIVE architecture, we propose a training scheme for supervised diarization systems. During training, when computing the loss of the voice activity detector, we remove the loss of frames that fall inside a collar from the total loss and backpropagate the resulting masked loss:

$$\mathcal{L}_{\text{vad}}^{\text{collar}}(\hat{y}, y) = -\frac{1}{TN}\sum_{\substack{t=1 \\ t\notin B_r}}^{T}\sum_{i=1}^{N}\log(\sigma(\hat{y}_{i,t}(2y_{i,t}-1))), \tag{7}$$

with B_r the set of frames that lie within a radius r around speaker turn boundaries. The effect of *collar-aware training* is illustrated in Figure 3. In Section 3.3, we show that training with the same collar as used for evaluation substantially improves the DER of the system. The total loss minimized by DIVE is therefore:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{selector}} + \mathcal{L}_{\text{vad}}^{\text{collar}}, \tag{8}$$

and is used to train jointly the temporal encoder, iterative speaker selector and voice activity detector.

3. EXPERIMENTS

We train our models on the "Fisher English Training Speech" Part 1 [35] and Part 2 [36], two datasets of conversational telephone speech. Since they contain clean sequences, we simulate noisy situations by adding background noise from the "noise" part of MUSAN [37]. More precisely, when sampling a training speech sequence, we also sample a random background noise. We renormalize the energy of both the speech and noise sequences, sample a gain uniformly in $[-20, 20]$ dB and apply it to the background noise before adding it to the speech sequence. We evaluate our models on the two-speaker evaluation of CALLHOME [27], a multilingual conversational speech dataset. Following [21, 25, 38] we report Diarization Error Rates averaged over the 148 test sequences. However, and unlike [21], we do not fine-tune our model on the 155 sequences of the "adaptation" set, but rather use it for hyperparameter selection. DERs are computed using the pyannote library [39].

3.1. Hyperparameters

The temporal encoder first reduces the length T of temporal embeddings with a 1D-Convolution with a kernel of size 16 and a stride of 8. It then cascades 4 blocks of 10 dilated convolution layers with kernel size 3 and stride 1. The dilation factor δ_l at layer l follows the pattern of [31, 24], i.e. $\delta_l = 2^{l \bmod 10}$, which means that we reinitialize the dilation factor at the beginning of each block. Between the first two blocks, we perform average pooling with kernel size 3 and stride 2. Thus, the total downsampling factor of the model is 16 ($T = L/16$). All convolutional layers use 512 feature maps. The two branches g_μ and g_h of the iterative speaker selector, as well as those (f_h and f_s) of the voice activity detector have two hidden layers with 512 feature maps.

We train our model with Adam [40] and a batch size of 512, using an initial learning rate of 0.0003, decayed by a factor of 0.7 every 50 k batches. We use multi-window training

Table 1. Diarization Error Rate (DER) in % on the test set of CALLHOME. All models are evaluated with a 250ms collar. "NO OVERLAP" means that the evaluation excludes overlapped speech.

Model	Overlap	No overlap
UIS-RNN V1 [41]	–	10.6
UIS-RNN V2 [41]	–	9.6
UIS-RNN V3 [41]	–	7.6
x-vector + LSTM [12]	–	6.6
BLSTM-EEND [17]	23.1	–
SA-EEND [21]	9.5	–
+ EDA [25]	8.1	–
+ Frame Selection [38]	7.8	–
DIVE	**6.7**	**5.9**

with 6 windows of length 32,000 samples each. Our ablation experiments are mostly performed with a smaller batch size (32) to lighten hardware requirements.

3.2. CALLHOME

Table 1 reports the DER on the test set of CALLHOME. The UIS-RNN [41] is an hybrid system training an RNN on top of pre-trained speaker embeddings, with the V3 being trained on a proprietary dataset with 138 k speakers. Similarly, [12] trains an LSTM to model the similarity between pre-trained speaker embeddings and performs diarization. Both models are evaluated without considering overlapped speech, and the latter uses oracle speech activity labels (removing silences). Table 1 shows that DIVE outperforms both systems in this condition, reaching 5.9% DER, even though DIVE is trained in an end-to-end fashion, without any speaker label and without oracle speech activity annotations. BLSTM-EEND [17] trains a bidirectional LSTM for per-speaker per-timestep voice activity detection, with an additional speaker clustering loss, similar in spirit to DIVE, with a DER of 23.1%. SA-EEND [21] replaces the LSTM by self-attention [20] and removes the deep clustering loss, with the best variation reaching 7.8% thanks to vast training data and fine-tuning on the CALLHOME validation set. DIVE outperforms these models, with a 6.7% DER, and despite not being fine-tuned on CALLHOME. In Table 1, the results for DIVE are obtained with a 11-frame median filtering on top of the model's predictions, as suggested in [17]. This avoids predicting non-existing, very short segments. Without this median filtering, the DER of DIVE goes up from 6.7% to 6.8%, which shows that the model's predictions are already reliable.

Table 2 analyzes error types. We observe few confusion errors where a speaker is mistaken for another (1.5%); most errors concentrate on mistaking single speaker activity for overlapped speech (7.4%), mistaking overlap for single speaker activity (3.3%) and mistaking silence for speaker activity (5.6%). Figure 2 plots the cumulative distribution of DER and shows that the median DER is below 5 while the average is higher due to a minority (5%) with DER over 20.

Table 2. Labels vs Predictions Contingency (%) for frame-wise diarization on CALLHOME.

Predictions	Labels			
	Spkr. 1	Spkr. 2	Overlap	Silence
Spkr. 1	**49.6**	0.9	1.8	3.5
Spkr. 2	0.6	**18.8**	1.5	2.1
Overlap	4.1	3.3	**8.4**	0.9
Silence	0.7	0.4	0.0	**3.3**
Class prior	55.1	23.3	11.8	9.8

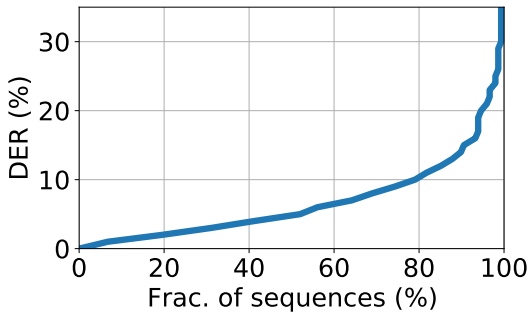

Fig. 2. Cumulative distribution of the Diarization Error Rate (DER) in % on CALLHOME, with the standard 250ms collar.

3.3. Impact of collar-aware training

Figure 3 shows the impact of collar-aware training. These results corresponds to our validation protocol (dev set results, training with a smaller batch size of 32), which is not directly comparable with our test protocol (Table 1). When using the standard loss function of Equation 6, the raw DER decreases steadily. On the other hand, when using the collar-aware loss defined in Equation 7, the raw DER plateaus early in training, but its DER with a 250ms collar converges faster and to a better score than its standard counterpart. This shows that when the target evaluation metric uses a collar, it is beneficial to integrate this tolerance into the training loss.

3.4. Impact of training on multiple windows

Diarization requires speaker representations that are reliable throughout long sequences of speech, e.g. several minutes. However, training a neural network over long speech sequences is slow since it prevents from training with a large batch size due to memory constraints. To solve this problem, our training process samples multiple short windows from the same sequence: this allows DIVE to observe the same speakers over snippets far apart in time while keep-

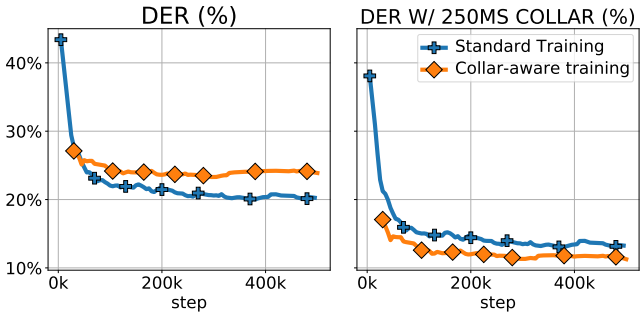

Fig. 3. Diarization Error Rate (DER) in % with and without collar-aware training, on the validation set of CALLHOME. On the left is the raw DER, that penalizes every error. On the right, the DER with the standard 250ms collar.

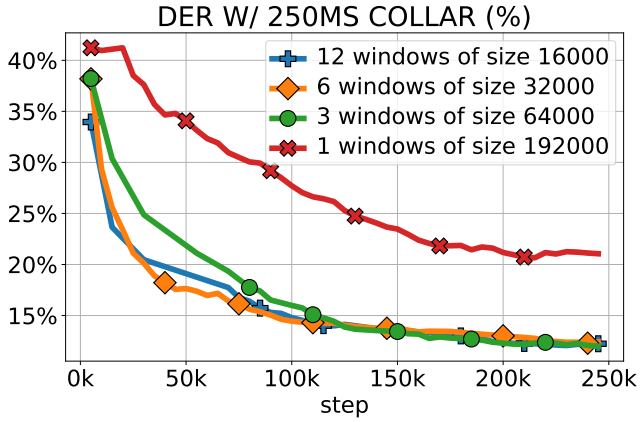

Fig. 4. Diarization Error Rate (DER) in % when varying the number and size of windows for a constant total of $192,000$ samples.

ing memory usage low. Figure 4 illustrates the benefit of multi-window training on our validation protocol: for a fixed budget of $192,000$ samples per training example, splitting it into several windows performs much better than using a single, contiguous window.

4. CONCLUSIONS

This paper introduces DIVE, an end-to-end model for speaker diarization. DIVE decomposes the task into three stages: convolutional temporal encoding, iterative speaker selection and speaker-conditioned voice activity prediction. The iterative speaker selector repeatedly processes the whole sequence to select a representation of a speaker not selected during the previous iterations. The extracted representations condition voice activity prediction. This formulation resolves the ambiguity in speaker order and offers a generic formulation regardless of the number of speakers per sequence. The model does not rely on pretrained components and all parameters are trained to optimize the voice activity likelihood with a novel collar-aware loss function. This loss does not rely on supervision from unreliable speaker turn boundaries, and matches standard collar-aware evaluations. DIVE establishes a new state-of-the-art on the standard CALLHOME benchmark, with 6.7% DER compared to 7.8% for the best alternative. In the future, we aim to address current limitations of DIVE. We will consider experimental settings with a variable number of speakers and noisier acoustic conditions [42, 43]. This will likely require improving the representation of already predicted speakers, as averaging speaker vectors may not provide a good representation in presence of many speakers.

5. ACKNOWLEDGEMENTS

Authors thank Raphaël Marinier for helpful discussions.

6. REFERENCES

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

[2] Xavier Anguera Miró, Simon Bozonnet, Nicholas W. D. Evans, Corinne Fredouille, Gerald Friedland, and Oriol Vinyals, "Speaker diarization: A review of recent research," *IEEE Trans. Speech Audio Process.*, vol. 20, no. 2, pp. 356–370, 2012.

[3] Tae Jin Park, Naoyuki Kanda, Dimitrios Dimitriadis, Kyu J. Han, Shinji Watanabe, and Shrikanth Narayanan, "A review of speaker diarization: Recent advances with deep learning," *CoRR*, vol. abs/2101.09624, 2021.

[4] Huazhong Ning, Ming Liu, Hao Tang, and Thomas S. Huang, "A spectral clustering approach to speaker diarization," in *INTERSPEECH 2006 - ICSLP, Ninth International Conference on Spoken Language Processing, Pittsburgh, PA, USA, September 17-21, 2006*. 2006, ISCA.

[5] Stephen Shum, Najim Dehak, Réda Dehak, and James R. Glass, "Unsupervised methods for speaker diarization: An integrated and iterative approach," *IEEE Trans. Speech Audio Process.*, vol. 21, no. 10, pp. 2015–2028, 2013.

[6] Stephen H Shum, Najim Dehak, Réda Dehak, and James R Glass, "Unsupervised methods for speaker diarization: An integrated and iterative approach," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 10, pp. 2015–2028, 2013.

[7] Gregory Sell and Daniel Garcia-Romero, "Speaker diarization with plda i-vector scoring and unsupervised calibration," in *2014 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2014, pp. 413–417.

[8] Dimitrios Dimitriadis and Petr Fousek, "Developing online speaker diarization system," in *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, Francisco Lacerda, Ed. 2017, pp. 2739–2743, ISCA.

[9] Dionyssos Kounades-Bastian, Laurent Girin, Xavier Alameda-Pineda, Sharon Gannot, and Radu Horaud, "An EM algorithm for joint source separation and diarisation of multichannel convolutive speech mixtures," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017*. 2017, pp. 16–20, IEEE.

[10] Daniel Garcia-Romero, David Snyder, Gregory Sell, Daniel Povey, and Alan McCree, "Speaker diarization using deep neural network embeddings," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 4930–4934.

[11] Quan Wang, Carlton Downey, Li Wan, Philip Andrew Mansfield, and Ignacio Lopz Moreno, "Speaker diarization with lstm," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5239–5243.

[12] Qingjian Lin, Ruiqing Yin, Ming Li, Hervé Bredin, and Claude Barras, "Lstm based similarity measurement with spectral clustering for speaker diarization," *arXiv preprint arXiv:1907.10393*, 2019.

[13] Keisuke Kinoshita, Marc Delcroix, and Naohiro Tawara, "Integrating end-to-end neural and clustering-based diarization: Getting the best of both worlds," *CoRR*, vol. abs/2010.13366, 2020.

[14] Qiujia Li, Florian L. Kreyssig, Chao Zhang, and Philip C. Woodland, "Discriminative neural clustering for speaker diarisation," in *IEEE Spoken Language Technology Workshop, SLT 2021, Shenzhen, China, January 19-22, 2021*. 2021, pp. 574–581, IEEE.

[15] Thad Hughes and Keir Mierle, "Recurrent neural networks for voice activity detection," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7378–7382.

[16] Samuel Thomas, George Saon, Maarten Van Segbroeck, and Shrikanth S. Narayanan, "Improvements to the IBM speech activity detection system for the DARPA RATS program," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015, South Brisbane, Queensland, Australia, April 19-24, 2015*. 2015, pp. 4500–4504, IEEE.

[17] Yusuke Fujita, Naoyuki Kanda, Shota Horiguchi, Kenji Nagamatsu, and Shinji Watanabe, "End-to-end neural speaker diarization with permutation-free objectives," *arXiv preprint arXiv:1909.05952*, 2019.

[18] Thilo von Neumann, Keisuke Kinoshita, Marc Delcroix, Shoko Araki, Tomohiro Nakatani, and Reinhold Haeb-Umbach, "All-neural online source separation, counting, and diarization for meeting analysis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 91–95.

[19] Yusuke Fujita, Shinji Watanabe, Shota Horiguchi, Yawen Xue, and Kenji Nagamatsu, "End-to-end neural diarization: Reformulating speaker diarization as simple multi-label classification," *arXiv preprint arXiv:2003.02966*, 2020.

[20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[21] Yusuke Fujita, Naoyuki Kanda, Shota Horiguchi, Yawen Xue, Kenji Nagamatsu, and Shinji Watanabe, "End-to-end neural speaker diarization with self-attention," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 296–303.

[22] Dong Yu, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 241–245.

[23] Yawen Xue, Shota Horiguchi, Yusuke Fujita, Shinji Watanabe, Paola García, and Kenji Nagamatsu, "Online end-to-end neural diarization with speaker-tracing buffer," in *IEEE Spoken Language Technology Workshop, SLT 2021, Shenzhen, China, January 19-22, 2021*. 2021, pp. 841–848, IEEE.

[24] Neil Zeghidour and David Grangier, "Wavesplit: End-to-end speech separation by speaker clustering," *arXiv preprint arXiv:2002.08933*, 2020.

[25] Shota Horiguchi, Yusuke Fujita, Shinji Watanabe, Yawen Xue, and Kenji Nagamatsu, "End-to-end speaker diarization for an unknown number of speakers with encoder-decoder based attractors," *arXiv preprint arXiv:2005.09921*, 2020.

[26] Yusuke Fujita, Shinji Watanabe, Shota Horiguchi, Yawen Xue, Jing Shi, and Kenji Nagamatsu, "Neural speaker diarization with speaker-wise chain rule," *CoRR*, vol. abs/2006.01796, 2020.

[27] NIST, "2000 speaker recognition evaluation plan," 2000.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*.

[29] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[30] Xavier Glorot, Antoine Bordes, and Yoshua Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, 2011, pp. 315–323.

[31] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio," in *SSW*. 2016, p. 125, ISCA.

[32] Jing Shi, Jiaming Xu, Yusuke Fujita, Shinji Watanabe, and Bo Xu, "Speaker-conditional chain model for speech separation and extraction," in *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, Helen Meng, Bo Xu, and Thomas Fang Zheng, Eds. 2020, pp. 2707–2711, ISCA.

[33] Thilo von Neumann, Keisuke Kinoshita, Marc Delcroix, Shoko Araki, Tomohiro Nakatani, and Reinhold Haeb-Umbach, "All-neural online source separation, counting, and diarization for meeting analysis," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019*. 2019, pp. 91–95, IEEE.

[34] Jing Shi, Xuankai Chang, Pengcheng Guo, Shinji Watanabe, Yusuke Fujita, Jiaming Xu, Bo Xu, and Lei Xie, "Sequence to multi-sequence learning via conditional chain mapping for mixture signals," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, Eds., 2020.

[35] Christopher Cieri, David Graff, Owen Kimball, Dave Miller, and Kevin Walker, "Fisher english training speech part 1," 2004.

[36] Christopher Cieri, David Graff, Owen Kimball, Dave Miller, and Kevin Walker, "Fisher english training speech part 2," 2005.

[37] David Snyder, Guoguo Chen, and Daniel Povey, "MUSAN: A Music, Speech, and Noise Corpus," Tech. Rep., 2015.

[38] Shota Horiguchi, Paola Garcia, Yusuke Fujita, Shinji Watanabe, and Kenji Nagamatsu, "End-to-end speaker diarization as post-processing," *arXiv preprint arXiv:2012.10055*, 2020.

[39] Hervé Bredin, "pyannote.metrics: a toolkit for reproducible evaluation, diagnostic, and error analysis of speaker diarization systems," in *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association*, Stockholm, Sweden, August 2017.

[40] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *ICLR (Poster)*, 2015.

[41] Aonan Zhang, Quan Wang, Zhenyao Zhu, John Paisley, and Chong Wang, "Fully supervised speaker diarization," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6301–6305.

[42] Neville Ryant, Prachi Singh, Venkat Krishnamohan, Rajat Varma, Kenneth Church, Christopher Cieri, Jun Du, Sriram Ganapathy, and Mark Liberman, "The third DIHARD diarization challenge," *CoRR*, vol. abs/2012.01477, 2020.

[43] Shinji Watanabe, Michael I. Mandel, Jon Barker, and Emmanuel Vincent, "Chime-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," *CoRR*, vol. abs/2004.09249, 2020.