Detecting and counting overlapping speakers in distant speech scenarios

Samuele Cornell, Maurizio Omologo, Stefano Squartini, Emmanuel Vincent

Detecting and Counting Overlapping Speakers in Distant Speech Scenarios

Samuele Cornell[⋆] *Maurizio Omologo* [†] *Stefano Squartini*[⋆] *Emmanuel Vincent*[¶]

[⋆] Department of Information Engineering, Università Politecnica delle Marche, Italy
[†] Center for Information and Communication Technology, Fondazione Bruno Kessler, Italy,
[¶] Université de Lorraine, CNRS, Inria, LORIA, France

s.cornell@pm.univpm.it, omologo@fbk.eu, s.squartini@univpm.it, emmanuel.vincent@inria.fr

Abstract

We consider the problem of detecting the activity and counting overlapping speakers in distant-microphone recordings. We treat supervised Voice Activity Detection (VAD), Overlapped Speech Detection (OSD), joint VAD+OSD, and speaker counting as instances of a general Overlapped Speech Detection and Counting (OSDC) task, and we design a Temporal Convolutional Network (TCN) based method to address it. We show that TCNs significantly outperform state-of-the-art methods on two real-world distant speech datasets. In particular our best architecture obtains, for OSD, 29.1% and 25.5% absolute improvement in Average Precision over previous techniques on, respectively, the AMI and CHiME-6 datasets. Furthermore, we find that generalization for joint VAD+OSD improves by using a speaker counting objective rather than a VAD+OSD objective. We also study the effectiveness of forced alignment based labeling and data augmentation, and show that both can improve OSD performance.

Index Terms: overlapped speech detection, speaker counting, distant speech, forced alignment

1. Introduction

Reliable multi-party speech diarization [1–4] and recognition [5–8] is still one of biggest challenges in the field of speech processing. It is well known that one of the main obstacles is the presence of overlapped speech which arises naturally in spontaneous human conversations. In fact, when encountered, speech recognition and diarization performance can degrade significantly. For this reason, overlapped speech detection (OSD) [9–11] is, along with voice activity detection (VAD) [12], one of the key components for any successful speech processing system [13, 14].

Supervised neural network based OSD systems have been shown to outperform more classical approaches [15–17]. Notably, Sajjan et al. [17] have shown that a neural network based approach for joint VAD+OSD can outperform a Gaussian Mixture Model (GMM) based system on real-world distant speech data such as the AMI meeting dataset [18]. Recent work focusing on OSD only [19, 20] has shown even more impressive results in near-field conditions.

In parallel, Stöter et al. [21] have shown that a neural network can be trained to estimate the number of concurrent speakers rather than simply performing joint VAD+OSD. This approach has been further expanded in [22] where three different output distributions for this speaker counting problem are proposed, different neural architectures are explored, and the performance is compared with humans. Also, in [23], a deep learning based speaker counting algorithm was evaluated against human ability and different single-channel features were compared. Crucially, these previous works on supervised speaker counting have relied only on close-talk, synthetic mixtures for training and testing.

Building on these previous works, we study supervised joint VAD+OSD and speaker counting in distant speech scenarios. We propose a Temporal Convolutional Network (TCN) architecture for these tasks, and evaluate it against previous works on joint VAD+OSD [17] and speaker counting [22] on AMI and CHiME-6 [14]. Because, to the best of our knowledge, supervised speaker counting has never been studied on real-world data, we also explore how the class imbalance problem can be mitigated using data augmentation. Finally, we investigate the use of forced alignment (FA) as a labeling procedure for front-end speech segmentation applications, as used in [17] on AMI and in CHiME-6 [14].

This paper is structured as follows. In Section 2, we briefly explain the multi-class classification framework adopted for joint supervised VAD+OSD and speaker counting and introduce our proposed TCN network architecture. We describe the datasets used in our experiments in Section 3 and present our study on FA-based labeling in Section 4. Finally, in Section 5, we report and discuss our experimental results and, in Section 6, we draw conclusions.

2. Proposed Overlapped Speech Detection and Counting Method

2.1. Overlapped Speech Detection and Counting Task

We treat supervised VAD, OSD, joint VAD+OSD, and speaker counting in a unified way, as instances of a general Overlapped Speech Detection and Counting (OSDC) task. This task can be formulated as a multi-class supervised learning problem, i.e., given a sequence of feature vectors $\mathbf{X} = \{\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n}\}$ and corresponding class labels $\mathbf{y} = \{y_1, y_2, \ldots, y_n\}$, we wish to find the optimal parameters $\boldsymbol{\theta}_{opt}$ of a model $\mathcal{F}(\mathbf{X}; \boldsymbol{\theta})$, which takes the feature vectors as inputs, and outputs the probability that the current frame belongs to a particular class.

In this framework, VAD and OSD can be treated either separately, as binary classification problems (speech vs. noise-only, overlap vs. non-overlap), or jointly, as a three-class (noise-only, single speaker, overlapped speech) problem. Speaker counting can be formulated as an $(N + 1)$-classes classification problem with N the maximum possible number of overlapping speakers [21]. While this approach is not the only one for supervised speaker counting, it has been found to be the most effective [22], provided the maximum possible number is known.

The work reported here was started at JSALT 2019, and supported by JHU with gifts from Amazon, Facebook, Google, and Microsoft.

2.2. TCN Architecture

We explore the use of a Temporal Convolutional Network (TCN) for OSDC, as this type of architecture has been shown to achieve state-of-the-art performance in many sequence-related tasks [24] and for source separation [25]. By employing stacked dilated convolutional layers, it exploits a long context and, at the same time, being fully convolutional, it is significantly faster than recurrent models in both training and inference phases.

Our architecture, depicted in Fig. 1, borrows some key design choices from [25, 26]. As input features, we use 80 log-mel filterbank features extracted from 25 ms windows with 10 ms stride. These are fed to a normalization layer followed by a 80×64 convolutional (conv) layer and by $R = 3$ blocks of $X = 5$ residual blocks (res blocks) with 1-D dilated convolutions, where in each block dilation factor increases as $2^0, 2^1, \ldots, 2^{X-1}$. Contrary to [25] we do not use skip connections as this degraded the performance for our task. Each residual block consists of a 64×128 point-wise convolution (conv 1×1), followed by normalization and activation, a dilated depth-wise separable 128×128 convolution (depth-conv), followed by normalization and activation, and another 128×64 point-wise convolution.

We use PReLU [27] as the activation function, layer normalization [28] for all normalization layers, and a kernel size of 3 in depth-wise dilated convolutions. A final $64 \times N$ point-wise convolution layer followed by softmax is used to output the probability of each frame belonging to one of the N classes (e.g., $N = 3$ for VAD+OSD). RAdam [29] is used for optimization and we tune the learning rate and the batch size for each experiment on a development set. Inference is performed over blocks of 600 frames with a stride of 300 frames. The outputs on overlapping blocks are averaged to obtain the final estimate.

The total number of parameters is 269k, and the whole algorithm, including feature extraction, runs 192 times faster than real-time on an i7-4771 processor.

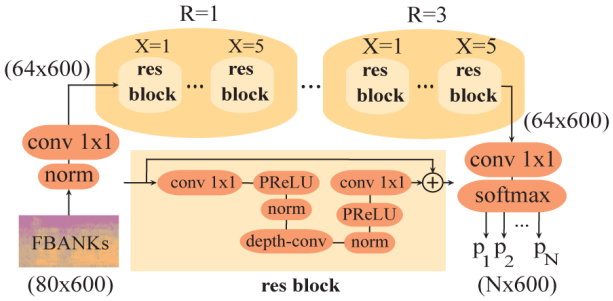

Figure 1: *Proposed TCN architecture for the OSDC task.*

3. Datasets

We conduct experiments on two real datasets: AMI and CHiME-6. In order to assess the impact of labeling error, we also build a controlled synthetic dataset. We describe them below.

3.1. Synthetic Dataset

Our synthetic dataset simulates multi-speaker recordings. Each simulated recording consists of clean speech utterances from Librispeech [30] *train-clean-100* convolved with artificial room impulse responses (RIRs) generated via gpuRIR [31] and contamined with noise from the CHiME-6 training set [14]. Each mixture involves 4 speakers, whose position and amount of overlap can change over time. For each mixture, we sampled randomly room size between $20 \div 50\ m^2$ and T60 reverberation time between $0.3 \div 0.7$ s. We simulated a close-talk reference microphone for each speaker. Thus, each RIR relates each speaker with a close-talk cardioid microphone input for a particular relative position. Ground truth word-level speaker activity was obtained from the clean utterances via the Montreal Forced Aligner (MFA) [32] and shifted according to the delay introduced by each RIR. We generated 10 synthetic mixtures, each with 19–23 min duration and with 20% percentage of overlapped speech using a total of 1663 RIRs.

3.2. AMI

The AMI Corpus [18] consists in over 100 h of meeting recordings. Each meeting has been recorded by a variety of devices including cameras, microphone arrays, and per-speaker headset and lapel microphones. Ground truth speaker activity was obtained by human annotators from close-talk worn microphones.

3.3. CHiME-6

The CHiME-6 corpus features more than 60 h of recordings organized in 20 sessions. Each session consists in a dinner party with 4 participants recorded with 6 microphone arrays. Due to the setting, it features highly conversational speech. Also, binaural microphones worn by each of the four participants were employed to provide close-talk references.

4. Labeling using Forced Alignment

We use our synthetic dataset to determine, in a controlled environment, whether FA can be considered as a reliable labeling procedure for the purpose of OSDC. In the past, FA-based labeling has been adopted in the CHiME-6 challenge for training and evaluation of diarization systems as well as in [17], with mixed results according to [19].

In Table 1 we report the statistics of word-level boundary errors resulting from FA, i.e., the difference in ms between the boundaries, estimated by a Kaldi [33] based Gaussian Mixture Model Hidden Markov Model (GMM-HMM) FA model applied on the simulated reference close-talk recordings, and the ground-truth boundaries, obtained with the MFA applied on the underlying clean utterances. We report the error percentiles achieved at the different training stages of the FA model from *mono* till *tri3* for two different simulations: one where the distance between each speaker and its reference microphone is fixed at 5 cm and another where is placed farther, at 60 cm. Since we expect errors to increase in the presence of overlapped speech, the percentiles are computed separately for overlapped vs. non-overlapped speech.

Except for the monophone model, FA exhibits small errors

Table 1: *Absolute error in ms of word-level boundaries estimated by FA on the synthetic dataset. The columns correspond to 40%, 60% and 80% percentiles. For each entry, the first and the second values correspond to 5 cm and 60 cm distance between each speaker and its reference microphone.*

Method	No Overlap [ms]			Overlap [ms]		
	80%	60%	40%	80%	60%	40%
mono	75/100	40/50	20/20	120/100	40/50	20/20
tri1	50/70	25/30	10/20	50/80	30/40	10/20
tri2	40/60	20/30	10/20	50/80	30/40	10/20
tri3	40/60	20/30	10/20	50/80	30/40	10/20

even when the reference microphone has been placed farther, most often below 30 ms and sometimes up to 50 ms, and the presence of overlapped speech does not significantly degrade the results. In the 60 cm case, greater errors up to 80 ms can be occasionally introduced in the presence of overlapped speech.

In the following, we will show experimentally on AMI, for which manual labels are available, that such errors do not impact the performance of a supervised speech segmentation algorithm. Conversely, using for training FA-based labels, obtained on per-speaker headset devices, improves the performance of the algorithm even when testing it on manually labeled data. This differs from the approach in [17] where FA-based labels were used both for training and testing, a procedure which could possibly be subject to bias in evaluation if not double-checked with manual annotation.

5. Experimental Evaluation

Hereafter, we present the results obtained for the proposed method (TCN) as well as the method in [17] (LSTM), originally proposed for joint VAD+OSD, and the best method in [22] (CRNN), originally proposed for speaker counting.[1] We choose [17] for comparison because, to the best of our knowledge, it is the only paper where supervised neural OSD has been applied to AMI far-field data. More recent works on supervised neural OSD such as [19, 20] only report results on AMI headset data.

The LSTM architecture has 2M parameters and outputs a prediction for every frame given a context of 11 frames. The CRNN has a lower parameter count of 157k and outputs a prediction for every frame given a context of 500 frames. The proposed method is significantly more computationally efficient than both.

For each architecture, we evaluate two versions: one trained for joint VAD+OSD and another one trained for speaker counting (COUNT). We are interested in comparing these two approaches because, on the one hand, speaker counting discriminates a larger set of classes but, on the other hand, it is plagued by the class imbalance problem which arises from conversational speech. The class imbalance can be seen in Table 2, which reports the class statistics for AMI and CHiME-6. For both datasets, the number of 4-speakers and 3-speakers frames is a small fraction of the total number of frames. Joint VAD+OSD is less informative than counting but possibly more robust, because the class imbalance appears to be more manageable.

We decided to report the performance of our algorithms using Average Precision (AP) which summarizes the Precision-Recall curve and is widely used, for example, in object segmentation [34] and other tasks where there is strong class imbalance. Scores are computed on 10 ms frame-level predictions. We also performed statistical significance testing for the values reported in Tables 3, 4, 5, 6, 7, 8 between top-3 performing architectures. For each Table, in each column, we highlight in bold font the best result and ones which are statistically equivalent to the best (if any) with $p = 0.001$. We choose Wilcoxon-Signed Rank non-parametric test [35] as we found metrics distribution to be highly non normal.

5.1. AMI

In our experiments on AMI we use as training material all channels from all microphone arrays as well as headset and lapel mixes. For testing, the first microphone of array 1 is used as in [17]. Unless stated otherwise, all scores in Tables 3, 4, and 5 are computed with respect to manual labels. Training is performed on FA-based labels instead obtained with a *tri3* Kaldi GMM-HMM model trained on speakers headset microphones.

To counteract the class imbalance problem, we use a data augmentation strategy similar to the one in [20] which we extend here also to speaker counting. We dynamically create at training time new examples for 2, 3, and 4 concurrent speakers by overlapping respectively 2, 3, and 4 random single-speaker chunks from the original dataset. To further increase training material, we normalize each chunk using a random gain factor sampled from $\mathcal{N}(\mu = -16.7, \sigma = 4)$ in dB scale. In this way, we augmented the original data by a factor of 70%. These hyperparameters were tuned on development-set. The impact of this augmentation scheme as well as the use of FA-based labels for training will be shown later in Section 5.3.

In Table 3, we report the AP scores achieved for VAD and OSD by the three architectures trained either for COUNT or for joint VAD+OSD. In the case of COUNT training, VAD and OSD are achieved at test time by summing the probabilities of the corresponding output classes. In Table 4, we report AP scores for each count (noise-only, 1 speaker, 2 speakers, etc.) for the architectures trained for COUNT.

Table 2: *Frame-level class frequency (%) for the AMI and CHiME-6 development and evaluation sets.*

Class frequency		noise	1-spk	2-spk	3-spk	4-spk
AMI	dev	15.87	67.17	13.95	2.59	0.42
	eval	15.12	68.39	12.63	3.09	0.76
CHiME-6	dev	24.05	54.25	17.74	3.49	0.47
	eval	33.47	51.52	12.03	2.46	0.51

Table 3: *AP (%) on the AMI development and evaluation sets for VAD and OSD.*

Method	VAD		OSD	
	dev	eval	dev	eval
TCN-COUNT	**98.5**	**98.4**	**63.2**	**56.6**
TCN-VAD+OSD	**98.5**	**98.5**	60.1	54.2
LSTM-COUNT	93.4	92.5	26.2	18.5
LSTM-VAD+OSD	93.3	92.8	26.7	19.2
CRNN-COUNT	94.1	93.2	33.4	27.5
CRNN-VAD+OSD	94.1	93.1	33.5	25.9

Table 4: *AP (%) on the AMI evaluation set for counting.*

Method	noise	1-spk	2-spk	3-spk	4-spk
TCN-COUNT	**50.7**	**86.1**	**40.4**	**11.3**	**0.03**
LSTM-COUNT	44.6	77.8	14.9	2.4	**0.02**
CRNN-COUNT	47.2	81.1	21.3	7.7	**0.03**

The proposed TCN-based architecture achieves the best figures for both joint VAD+OSD and COUNT variants. More generally, as we could expect, the AP score for VAD is remarkably higher than for OSD (Table 3). This suggests that OSD is a more challenging task, but also that the data augmentation scheme is not enough to counteract the class imbalance. This difference is even more extreme when looking at the counting performance (Table 4), where high AP values are obtained only for noise-only and 1-speaker classes and all architectures fail to give reliable predictions for 3 and 4 speakers.

[1] Code for synthetic data and experiments is available at github.com/popcornell/OSDC.

Table 5: *Accuracy (%) on the AMI evaluation set compared with [17]. The scores obtained in [17] using their FA-based labels are reported in parentheses.*

Method	VAD	OSD
TCN-COUNT	**94.0**	**95.4**
TCN-VAD+OSD	**94.2**	94.6
LSTM-VAD+OSD (re-implemented)	93.3	91.52
LSTM-VAD+OSD [17]	77 (71.0)	68 (87.9)

An unexpected result is that the TCN architecture trained for COUNT outperforms the one trained for joint VAD+OSD when evaluated on the OSD task, while both training objectives result in similar performance on the VAD task.

In Table 5, we further compare the proposed approach as well as our re-implementation of [17] with the results originally reported in terms of accuracy in [17] on the AMI evaluation set. Both our re-implementation and our proposed method reach higher accuracy for both VAD and OSD. Having re-implemented the same architecture (LSTM-VAD+OSD), the improvement over the original [17] is mainly due to the fact that we use a better data augmentation strategy while [17] used out-of-domain synthetic data. FA-obtained labels, instead, are used both in [17] and in this work. It must be noted, however, that accuracy does not fully reflect performance for OSD. In fact, by supposing no overlapped speech in all evaluation set, an accuracy of 83% is obtained.

5.2. CHiME-6

Hereafter we present the results obtained on the CHiME-6 corpus. Training was performed using all microphone channels from all array devices with FA-based labels obtained using the official challenge baseline Kaldi recipe[2]. For development and evaluation, we used the provided official FA-based labels as ground truth and we averaged predictions across all microphone channels. Regarding data-augmentation the same strategy adopted for AMI was used but with an augmentation factor of 40%.

Tables 6 and 7 report the AP achieved by all architectures for VAD and OSD and for counting, respectively. Compared to AMI, we obtain lower AP values in general. In fact, due to its less constrained setting, CHiME-6 is arguably a much more challenging dataset.Nevertheless, the trends observed on AMI are preserved: the TCN architecture consistently performs better, and the AP for OSD remains much lower than the one for VAD. This is reflected in the lower AP values obtained for 2, 3, and 4 speakers.

Again, the TCN architecture trained for COUNT outperforms the one trained for joint VAD+OSD for the OSD task on the evaluation set, and performs similarly for VAD. This result is in accordance to what has been found on AMI. Teaching the network to count speakers, rather than merely identifying the presence of overlapped speech, leads the model to perform better. In this particular case, speaker counting helps generalization as the network is forced to learn a more difficult task.

5.3. Ablation Study

Table 8 compares the results obtained by our best performing method (TCN-COUNT) with those obtained by the same TCN architecture without data augmentation and without FA-based labeling (in the latter case, manual labels are used for training instead). We can see that performance drops significantly without FA labeling especially on AMI, possibly because, for this dataset, manual annotation is less reliable as explained in [19]. Data-augmentation improves the performance on AMI but gives mixed results on CHiME-6. In fact while it slightly improves OSD AP the VAD AP degrades. We hypothesize this is due to the fact that CHiME-6 is considerably more noisy and by overlapping multiple speech segments also the noise adds up, resulting in less realistic synthetic mixtures especially in non-speech regions.

[2]FA-based labels were provided for the development and evaluation sets only.

Table 6: *AP (%) on the CHiME-6 development and evaluation sets for VAD and OSD.*

Method	VAD		OSD	
	dev	eval	dev	eval
TCN-COUNT	**93.5**	94.2	55.8	**51.1**
TCN-VAD+OSD	**93.4**	**94.4**	**56.1**	49.1
LSTM-COUNT	91.8	89.4	20.1	17.4
LSTM-VAD+OSD	92.5	91.2	20.4	17.1
CRNN-COUNT	94.1	93.2	29.2	25.7
CRNN-VAD+OSD	94.3	93.4	27.8	21.3

Table 7: *AP (%) on the CHiME-6 evaluation set for counting.*

Method	noise	1-spk	2-spk	3-spk	4-spk
TCN-COUNT	**89.8**	**77.6**	**31.1**	**12.1**	**0.003**
LSTM-COUNT	71.6	68.8	11.3	3.2	**0.002**
CRNN-COUNT	79.3	72.5	13.4	6.2	**0.003**

Table 8: *AP (%) achieved by TCN-COUNT on the AMI and CHiME-6 evaluation sets for VAD and OSD without data augmentation and without FA-based labels.*

TCN-COUNT	VAD		OSD	
	AMI	CHiME-6	AMI	CHiME-6
Full	**98.4**	94.2	**56.6**	**51.1**
w/o Augm	96.6	**94.5**	40.7	48.3
w/o FA-labels	74.3	91.5	33.0	48.1

6. Conclusions

In this work we studied the problem of activity detection of multiple speakers with far-field microphones. Specifically, we focused and compared two different approaches: joint VAD+OSD and speaker counting which, to the best of our knowledge, has never been studied on real-world data. For both approaches we proposed a TCN-based neural network architecture which was shown to outperform previously proposed architectures on AMI and CHiME-6. As an additional contribution we also studied the use of forced-alignment as a labeling procedure for these tasks and we showed that it can improve performance even when manual annotation is used for evaluation.

In our experiments we found that the class imbalance problem which plagues speaker counting and, to a less extent, OSD can be partially mitigated by using data augmentation but both remain very challenging tasks in real-world distant speech scenarios. In particular, while we failed at training a reliable speaker counting system, we showed that it can be more convenient to use a speaker counting objective to perform joint VAD+OSD rather than training a network directly with a joint

VAD+OSD objective, as the network trained to count shows improved generalization.

7. References

[1] N. Ryant, E. Bergelson, K. Church, A. Cristia, J. Du, S. Ganapathy, S. Khudanpur, D. Kowalski, M. Krishnamoorthy, R. Kulshreshta, M. Liberman, Y. Lu, M. Maciejewski, F. Metze, J. Profant, L. Sun, Y. Tsao, and Z. Yu, "Enhancement and analysis of conversational speech: JSALT 2017," in *ICASSP*, 2018, pp. 5154–5158.

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

[3] M. Diez, F. Landini, L. Burget, J. Rohdin, A. Silnova, K. Žmolíková, O. Novotný, K. Veselý, O. Glembek, O. Plchot, L. Mošner, and P. Matějka, "BUT system for DIHARD speech diarization challenge 2018," in *Interspeech*, 2018, pp. 2798–2802.

[4] 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.

[5] J. Li, L. Deng, R. Haeb-Umbach, and Y. Gong, *Robust Automatic Speech Recognition — A Bridge to Practical Applications*. Elsevier, 2015.

[6] S. Watanabe, M. Delcroix, F. Metze, and J. R. Hershey, Eds., *New Era for Robust Speech Recognition — Exploiting Deep Learning*. Springer, 2017.

[7] E. Vincent, T. Virtanen, and S. Gannot, Eds., *Audio Source Separation and Speech Enhancement*. Wiley, 2018.

[8] R. Haeb-Umbach, S. Watanabe, T. Nakatani, M. Bacchiani, B. Hoffmeister, M. L. Seltzer, H. Zen, and M. Souden, "Speech processing for digital home assistants: Combining signal processing with deep-learning techniques," *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 111–124, 2019.

[9] K. Boakye, B. Trueba-Hornero, O. Vinyals, and G. Friedland, "Overlapped speech detection for improved speaker diarization in multiparty meetings," in *ICASSP*, 2008, pp. 4353–4356.

[10] K. Boakye, O. Vinyals, and G. Friedland, "Two's a crowd: Improving speaker diarization by automatically identifying and excluding overlapped speech," in *Interspeech*, 2008, pp. 32–35.

[11] S. Otterson and M. Ostendorf, "Efficient use of overlap information in speaker diarization," in *ASRU*, 2007, pp. 683–686.

[12] J. Ramırez, J. C. Segura, C. Benıtez, A. De La Torre, and A. Rubio, "Efficient voice activity detection algorithms using long-term speech information," *Speech Communication*, vol. 42, no. 3-4, pp. 271–287, 2004.

[13] L. P. García-Perera, J. Villalba, H. Bredin, J. Du, D. Castán, A. Cristia, L. Bullock, L. Guo, K. Okabe, P. S. Nidadavolu, S. Kataria, S. Chen, L. Galmant, M. Lavechin, L. Sun, M.-P. Gill, B. Ben-Yair, S. Abdoli, X. Wang, W. Bouaziz, H. Titeux, E. Dupoux, K. A. Lee, and N. Dehak, "Speaker detection in the wild: Lessons learned from JSALT 2019," in *Odyssey*, 2020, pp. 415–422.

[14] S. Watanabe, M. Mandel, J. Barker, E. Vincent, A. Arora, X. Chang, S. Khudanpur, V. Manohar, D. Povey, D. Raj, D. Snyder, A. S. Subramanian, J. Trmal, B. B. Yair, C. Boeddeker, Z. Ni, Y. Fujita, S. Horiguchi, N. Kanda, T. Yoshioka, and N. Ryant, "CHiME-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," in *CHiME*, 2020.

[15] J. Geiger, F. Eyben, B. Schuller, and G. Rigoll, "Detecting overlapping speech with long short-term memory recurrent neural networks," in *Interspeech*, 2013, pp. 1668–1672.

[16] V. Andrei, H. Cucu, and C. Burileanu, "Detecting overlapped speech on short timeframes using deep learning," in *Interspeech*, 2017, pp. 1198–1202.

[17] N. Sajjan, S. Ganesh, N. Sharma, S. Ganapathy, and N. Ryant, "Leveraging LSTM models for overlap detection in multi-party meetings," in *ICASSP*, 2018, pp. 5249–5253.

[18] I. Mccowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska Masson, W. Post, D. Reidsma, and P. Wellner, "The AMI meeting corpus," in *5th International Conference on Methods and Techniques in Behavioral Research*, 2005, pp. 137–140.

[19] M. Kunešová, M. Hrúz, Z. Zajíc, and V. Radová, "Detection of overlapping speech for the purposes of speaker diarization," in *International Conference on Speech and Computer*, 2019, pp. 247–257.

[20] L. Bullock, H. Bredin, and L. P. Garcia-Perera, "Overlap-aware diarization: Resegmentation using neural end-to-end overlapped speech detection," in *ICASSP*, 2020, pp. 7114–7118.

[21] F. Stöter, S. Chakrabarty, B. Edler, and E. Habets, "Classification vs. regression in supervised learning for single channel speaker count estimation," in *ICASSP*, 2018, pp. 436–440.

[22] F. Stöter, S. Chakrabarty, B. Edler, and E. A. P. Habets, "Countnet: Estimating the number of concurrent speakers using supervised learning," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 27, no. 2, pp. 268–282, 2019.

[23] V. Andrei, H. Cucu, and C. Burileanu, "Overlapped speech detection and competing speaker counting — humans versus deep learning," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 4, pp. 850–862, 2019.

[24] S. Bai, J. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint:1803.01271*, 2018.

[25] Y. Luo and N. Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[26] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1026–1034.

[28] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *Stat*, vol. 1050, p. 21, 2016.

[29] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the variance of the adaptive learning rate and beyond," in *International Conference on Learning Representations*, 2020.

[30] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *ICASSP*, 2015, pp. 5206–5210.

[31] D. Diaz-Guerra, A. Miguel, and J. R. Beltran, "gpuRIR: A Python library for room impulse response simulation with GPU acceleration," *arXiv preprint:1810.11359*, 2018.

[32] M. McAuliffe, M. Socolof, S. Mihuc, M. Wagner, and M. Sonderegger, "Montreal Forced Aligner: Trainable text-speech alignment using Kaldi," in *Interspeech*, 2017, pp. 498–502.

[33] D. Povey, A. Ghoshal *et al.*, "The Kaldi speech recognition toolkit," in *ASRU*, 2011.

[34] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European Conference on Computer Vision (ECCV)*, 2014, pp. 740–755.

[35] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, no. Jan, pp. 1–30, 2006.