Property-Aware Multi-Speaker Data Simulation: A Probabilistic Modelling Technique for Synthetic Data Generation

Tae Jin Park, He Huang, Coleman Hooper, Nithin Koluguri, Kunal Dhawan, Ante Jukić, Jagadeesh Balam and Boris Ginsburg

NVIDIA, Santa Clara, USA

{taejinp,heh,chooper,nkoluguri,kdhawan,ajukic,jbalam,bginsburg}@nvidia.com

Abstract

We introduce a sophisticated multi-speaker speech data simulator, specifically engineered to generate multi-speaker speech recordings. A notable feature of this simulator is its capacity to modulate the distribution of silence and overlap via the adjustment of statistical parameters. This capability offers a tailored training environment for developing neural models suited for speaker diarization and voice activity detection. The acquisition of substantial datasets for speaker diarization often presents a significant challenge, particularly in multi-speaker scenarios. Furthermore, the precise time stamp annotation of speech data is a critical factor for training both speaker diarization and voice activity detection. Our proposed multi-speaker simulator tackles these problems by generating large-scale audio mixtures that maintain statistical properties closely aligned with the input parameters. We demonstrate that the proposed multi-speaker simulator generates audio mixtures with statistical properties that closely align with the input parameters derived from real-world statistics. Additionally, we present the effectiveness of speaker diarization and voice activity detection models, which have been trained exclusively on the generated simulated datasets.

Index Terms: speaker diarization, data simulator, multi-speaker data simulation

1. Introduction

The evolution of deep neural network models within the realm of speech signal processing has significantly enhanced the performance and precision of the machine learning systems [1]. These advances have facilitated an end-to-end training approach, allowing the entire model to be optimized, transforming raw audio input into meaningful labels. However, achieving competitive accuracies with these neural models depends on the procurement of a substantial amount of data. This data is integral to ensuring generalizability and improving accuracy.

Obtaining sufficient training data in certain domains poses a significant challenge due to an array of factors. In speech signal processing field, the challenges are concentrated on privacy concerns, data-imbalance issues, limited availability and the financial cost of data collection. The task becomes even more demanding when it involves speaker diarization. This increased difficulty is primarily because speaker diarization requires a complex dataset with multiple speakers, embodying a broad range of variabilities. These variabilities encompass aspects such as gender, acoustic conditions, and conversation types. Hence, the development and optimization of effective deep neural network models for speech signal processing, particularly speaker diarization, hinges on overcoming these challenges related to data collection.

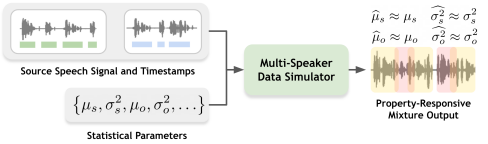


Figure 1: *Property-aware multispeaker data simulator that generates targeted amount of pause and overlap.*

In response to the challenge of data scarcity in specific fields, the machine learning community has adopted synthetic data, which mitigates the aforementioned issues to a certain degree. In order for synthetic data to be effective, the generated data should capture the characteristics and patterns of real-world data (*realism*) while maintaining a broad range of variations (*diversity*). Also, accurate and consistent labeling of the synthetic dataset is essential. Additionally, in speaker diarization or Voice Activity Detection (VAD), the diversity of speakers, sentence length, and frequency of speaker turns in conversations should be well balanced, mirroring real-world data.

Until now, in the fields of speech recognition and speaker diarization, most published articles have focused on data augmentation techniques, such as the widely used SpecAugment [2] or the data augmentation speech recognition toolkit [3, 4]. There exist simulation tools (e.g., one featured in [5]) initially developed for source separation but often utilized in training speaker diarization systems [5, 6, 7]. Recently, a multi-speaker data simulator for speaker end-to-end speaker diarization also appeared in [8, 9], which tries to create mixtures that resemble the pauses and overlaps of the real-world audio recordings. While the data simulation techniques introduced in [6, 9] serve their purpose very well, these data simulation techniques tend to employ a range of parameters which do not explicitly correlate with specific properties such as pauses and overlaps within the resulting simulated speech recordings. Consequently, even though the previously proposed simulation systems accept numerous parameters, their lack of control over the generated signal could lead to unpredictability in the amount of silence and overlap.

In this work, we introduce a dynamic sampling technique that constantly reflects the discrepancy between the generated data and the targeted amount of overlap speech and silence employing probabilistic models for precision and control. We refer to such feature as *"Property-aware simulation"*. As illustrated in Fig. 1, the proposed multi-speaker data simulator takes speech signal and its alignment (time stamps) and blends these signals to simulate multi-speaker audio recordings. Herein, we elaborate on the guiding principles for our data simulation systems:

- The simulated sessions are designed to incorporate the required amount of silence, overlap, and sentence length based on statistical analysis.

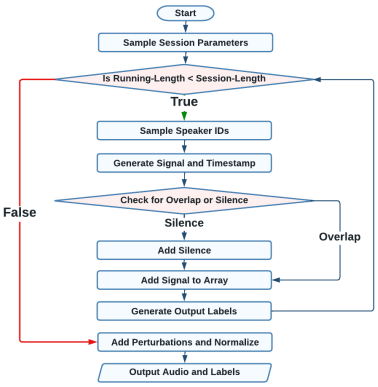


Figure 2: *Flowchart of the proposed multi-speaker data simulator*

- The speech signal generated by the simulation system exhibits a significant level of variability across sessions, including overlap ratio, silence ratio, and average sentence length.
- The simulation system employs parallel processing techniques, leveraging multiple graphics processing units (GPUs), enabling large-scale data generation at higher speed.
- The implementation of the data simulator is open-source and publicly available online.[1]

2. System description

2.1. Major parameters

A flow diagram of the proposed system is shown in Fig. 2. In the following sections we describe the main parameters and implementation details. Note that the following parameters are the most crucial subset of parameters that are determined before starting data simulation:

- Session length L_S: A floating point number that determines the total duration of the created session in second.
- Number of sessions N_S: An integer to determine the number of session to be simulated, so that the total duration of the generated data is $L_S \cdot N_S$ seconds.
- Number of speakers N_{spk}: An integer number that determines how many speakers in a session.
- Turn Probability p_{turn}: A floating point number that determines the speaker turn change from one to another.
- Overlap ratio mean μ_o and variance σ_o^2: Parameters that determine the distribution of overlap.
- Silence ratio mean μ_s and variance σ_s^2: Parameters that determine the distribution of silence.

The following are random variables that are created at each session:

- Sentence length s_l determines how many words are included in a newly added utterance (also referred to as a sentence).
- Silence length $\widetilde{m}_s$ determines the duration between sentences.
- Overlap length $\widetilde{m}_o$ determines how much portion of speech is overlapped with the following speech segment.

2.2. Session Parameter Sampling

The following variables are sampled during the very first step named "*Sample Session Parameters*" in Fig. 2.

- Session random seed sampling: Set a random seed value which would be used to create a reproducible simulation environment.
- Set speaker dominance: Call a method that randomly determines the dominance of each speaker in the session.
- Speaker volumes: Set the volume level of each speaker in the session.
- Setting Session Silence and Overlap Mean: To control the amount of silence and overlap in a session, we can set the mean values for these parameters using the following equation, which describes the method of moment estimates [10] for a Beta distribution[2] :

$$\alpha_{\{o,s\}} = \frac{\mu_{\{o,s\}}^2 \cdot (1 - \mu_{\{o,s\}})}{\sigma_{\{o,s\}}^2} - \mu_{\{o,s\}} \quad (1)$$

$$\beta_{\{o,s\}} = \frac{\mu_{\{o,s\}} \cdot (1 - \mu_{\{o,s\}})^2}{\sigma_{\{o,s\}}^2} - (1 - \mu_{\{o,s\}}) \quad (2)$$

Here, μ represents the mean ratio of silence or overlap and σ represents its variance. These parameters are fed by the user to control the overall silence and overlap ratio. To ensure that $\alpha_{\{o,s\}}>0$ and $\beta_{\{o,s\}}>0$, the input mean and variance values should be within the following range:

$$\begin{cases} 0 < \mu_{\{o,s\}} < 1 \\ 0 < \sigma_{\{o,s\}}^2 \leq \mu_{\{o,s\}}(1 - \mu_{\{o,s\}}) \end{cases} \quad (3)$$

We can then sample the session silence mean X_{μ_s} and session overlap mean X_{μ_o} from the Beta distribution, as follows:

$$X_{\mu_s} \sim Beta(\alpha_s, \beta_s), \quad (4)$$

$$X_{\mu_o} \sim Beta(\alpha_o, \beta_o). \quad (5)$$

Here, α_s and β_s are based on the mean μ_s and variance σ_s for the silence ratio in a session, while α_o and β_o are based on the mean μ_o and variance σ_o for the overlap speech ratio in a session. By setting the session silence and overlap mean values in this way, we can control the amount of silence and overlap in a session, which follows Beta distribution.

2.3. Sampling Routine for Data simulation

The following provides a description of each step involved in generating a simulated multi-speaker audio recording. In this section: n_s denotes the current sample count, s_{spk} the speaker index, and $\tilde{L}_S$ the running length of the audio signal thus far.

2.3.1. Data synthesis loop

As described in the Algorithm 1, the running length of the current session $\tilde{L}_S$ is monitored at every loop and while the condition $\tilde{L}_S < L_S$ is held, the sampling process is continued until the running length $\tilde{L}_S$ exceeds the desired length L_S.

2.3.2. Sample Speaker ID

The turn probability p_{turn} is compared with a value drawn from a uniform distribution.

$$U(0, 1) < p_{\text{turn}} \quad (6)$$

If the sampled value is less than the p_{turn} value, a randomly chosen speaker is selected from the pre-determined speaker group, for example $\mathcal{S}_{spks} = \{s_1, s_2, \ldots, s_{N_{spk}}\}$.

[1] https://github.com/NVIDIA/NeMo/main/tools/speech_data_simulator

[2] Beta distribution is employed due to its compatibility with the range of overlap and silence ratios, which fall within its support of [0, 1], and its capacity to model skewed distributions[11].

2.3.3. Build Sentence

The parameter s_l, which represents sentence length, is assumed to follow a negative binomial distribution. This approach is based on the probabilistic model for word-level sentence length (measured in words) of human language as detailed in [12].

$$s_l \sim \text{NB}(k_w, p_w) \tag{7}$$

$$P_{\text{NB}}(X = k_w) = \binom{X + k_w - 1}{k_w - 1} p_w^{k_w} (1 - p_w)^X \tag{8}$$

Based on the sentence length s_l and speaker (also referred as speaker turn) s_{spk}, we randomly select the given number of words from the forced-alignment data. This process is denoted as BUILDSENTENCE() function in the Algorithm 1.

$$\tilde{L}_{spch}, \tilde{L}_{sil} = \text{BUILDSENTENCE}(s_l, s_{spk}). \tag{9}$$

2.3.4. Overlap-Silence Selector

In this step, the data-simulator system compares the current silence ratio to the current overlap ratio. Thus, at each utterance loop, it switches to either silence or overlap mode according to the amount of the gap between current ratio and session mean in configurations.

$$\Delta S = \frac{\tilde{L}_{sil}}{\tilde{L}_S} - \mu_s \tag{10}$$

$$\Delta O = \frac{\tilde{O}_{spch}}{\tilde{L}_{spch}} - \mu_o \tag{11}$$

We employ two different quantities: *silence discrepancy* ΔS represents the gap between desired silence time and the current silence time and *overlap discrepancy* ΔO represents which is the gap between desired overlap speech and the current overlap speech time. We choose whichever is smaller than other.

2.3.5. Estimating the Required Overlap Amount

Overlap $\widetilde{m}_o$ is calculated so that the newly added amount of overlap matches the expected amount of X_{μ_o}.

$$X_{\mu_o} = \frac{\widetilde{m}_o + \tilde{O}_{spch}}{\tilde{L}_{spch} - \widetilde{m}_o} \tag{12}$$

Afterwards, we solve for $\widetilde{m}_o$ and assign it the value derived from the following equation:

$$\widetilde{m}_o \leftarrow \frac{X_{\mu_o}\tilde{L}_{spch} - \tilde{O}_{spch}}{X_{\mu_o} + 1} \tag{13}$$

2.3.6. Estimating the Required Silence Amount

We set up an equation that matches the expected amount of silence after adding the silence (denoted by $\widetilde{m}_s$) with the sampled mean X_{μ_s} as follows:

$$X_{\mu_s} = \frac{\widetilde{m}_s + \tilde{L}_{sil}}{\widetilde{m}_s + \tilde{L}_S} \tag{14}$$

Solve for $\widetilde{m}_s$ then we assign the following value:

$$\widetilde{m}_s \leftarrow \frac{\tilde{L}_{sil} - X_{\mu_s}\tilde{L}_S}{X_{\mu_s} - 1} \tag{15}$$

Algorithm 1 Dialogue Simulation

Require: $L_S, \sigma_d^2\ \mu_o, \mu_s, p_{\text{turn}}\ \sigma_o^2, \sigma_s^2 \in \mathbb{R}$ and $N_{spk} \in \mathbb{N}$
$p \in (0, 1] \vee \mu_d, \mu_o, \mu_s, \sigma_d^2, \sigma_o^2, \sigma_s^2 \in [0, 1]$
$(\alpha_s, \beta_s) \leftarrow \left(\mu_s^2 \frac{(1-\mu_s)}{\sigma_s^2} - \mu_s, \mu_s \frac{(1-\mu_s)^2}{\sigma_s^2} - (1 - \mu_s)\right)$
$X_{\mu_s} \sim Beta(\alpha_s, \beta_s)$ ▷ Sample session silence rate mean
$(\alpha_o, \beta_o) \leftarrow \left(\mu_o^2 \frac{(1-\mu_o)}{\sigma_o^2} - \mu_o, \mu_o \frac{(1-\mu_o)^2}{\sigma_o^2} - (1 - \mu_o)\right)$
$X_{\mu_o} \sim Beta(\alpha_o, \beta_o)$ ▷ Sample session overlap rate mean
while $\tilde{L}_S < L_S$ **do**
  if $U(0, 1) < p_{\text{turn}}$ **then**
    s_{spk} = GETNEXTSPEAKER($\mathcal{S}_{spks}, s_{spk}$)
  end if
  $s_l \sim NB(k_w, p_w)$
  $\tilde{L}_{spch}, \tilde{L}_{sil} \leftarrow$ BUILDSENTENCE(s_l, s_{spk})
  $\Delta S \leftarrow \frac{\tilde{L}_{sil}}{\tilde{L}_S} - \mu_s$ ▷ Silence deficiency
  $\Delta O \leftarrow \frac{\tilde{O}_{spch}}{\tilde{L}_{spch}} - \mu_o$ ▷ Overlap deficiency
  if $\Delta S \leq \Delta O$ **then**
    $\widetilde{m}_s \leftarrow \frac{\tilde{L}_{sil} - X_{\mu_s}\tilde{L}_S}{X_{\mu_s} - 1}$
    $k_s \leftarrow \widetilde{m}_s^2/\sigma_s^2$
    $\theta_s \leftarrow \sigma_s^2/\widetilde{m}_s$
    $s_{\Delta t} \sim \Gamma(k_s, \theta_s)$
    ADDSENTENCE($s_{\Delta t}$, 0)
  else if $\Delta S > \Delta O$ **then**
    $\widetilde{m}_o \leftarrow \frac{X_{\mu_o}\tilde{L}_{spch} - \tilde{O}_{spch}}{X_{\mu_o} + 1}$
    $k_o \leftarrow \widetilde{m}_o^2/\sigma_o^2$
    $\theta_o \leftarrow \sigma_o^2/\widetilde{m}_o$
    $o_{\Delta t} \sim \Gamma(k_o, \theta_o)$
    ADDSENTENCE(0, $o_{\Delta t}$)
  end if
end while

2.3.7. Sampling overlap and silence amount

For both silence and overlap cases, we employ gamma distribution since gamma distribution is continuous version of negative binomial distribution that is used to model sentence length in [12] Thus, we model the distribution of the two continuous quantity, silence and overlap length, as following equations:

$$k \leftarrow \widetilde{m}^2/\sigma^2 \tag{16}$$

$$\theta \leftarrow \sigma^2/\widetilde{m} \tag{17}$$

$$x_{\Delta t} \sim \Gamma(k, \theta), \tag{18}$$

where $x_{\Delta t}$ is the sampled silence amount $s_{\Delta t}$ in silence case and overlap amount $o_{\Delta t}$ in overlap case.

3. Experimental Results

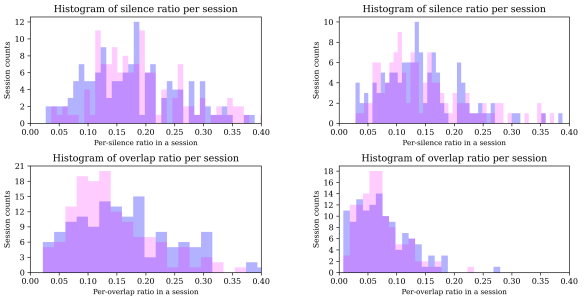
3.1. Data simulation test

In this section, we test whether the proposed data simulator can simulate the multi speaker data with the given parameters. To check wether the simulator generates data which has the distribution we intended to create, we compare the simulated data with the statistics extracted from real-world datasets. The overlap mean, overlap mean variance, silence mean and silence mean var values are collected from the real-world datasets and fed to the simulator.

Table 1 presents a quantitative comparison of observed values derived from both the simulated and real-world datasets for *train* split of AMI(MixHeadSet) [13] and CallHome American English Speech (CHAES) [14], highlighting key metrics such

Table 1: *Simulated vs. Real-world Dataset Statistics*

Dataset	Type	Mean	Var.
CH. Simul.	observed sil. ratio	0.1409	0.0045
CHAES	real-world sil. ratio	0.1473	0.0061
CH. Simul.	observed ovl. ratio	0.0759	0.0019
CHAES	real-world ovl. ratio	0.0754	0.0020
AMI Simul.	observed sil. ratio	0.1804	0.0077
AMI	real-world sil. ratio	0.1814	0.0081
AMI. Simul.	observed ovl. ratio	0.1711	0.0092
AMI	real-world ovl. ratio	0.1473	0.0047



(a) *AMI-train 139 sessions* (b) *CH109 - 109 sessions*

Figure 3: *Histograms: Real-World (Magenta) vs. Simulated (Blue) Data; Overlaps in Purple.*

as the mean and variance of silence (sil.) and overlap (ovl.) ratios. Notwithstanding certain disparities between the statistics discerned from the simulated dataset and those from the real-world dataset, the simulation effectively echoes the trends characteristic of the original statistics. For an expanded analysis of this simulation's distribution, please refer to Fig. 3, which exhibits histograms contrasting the original and simulated datasets in terms of overlap and silence mean/variance.

3.2. Voice Activity Detector Model

We trained a modified version of the Voice Activity Detection (VAD) model [3] proposed in [15], using our simulated data. As source datasets, we employed Fisher English Corpus [16] and LibriSpeech Corpus[17]. For Fisher dataset, we use energy based VAD to filter out salient speech samples and randomly segmented audio in a range of [0.2, 0.8] seconds word length. For LibriSpeech, we use the forced alignment result in [18]. We utilize two datasets: Dataset D1 comprises 0.5k hours of data from each of the LibriSpeech and Fisher English datasets. Dataset D2 consists of 1k hours from each of the LibriSpeech and Fisher datasets, supplemented by an additional 2.5k hours of multilingual data we have gathered from [19, 20, 21, 22].

The performance of this modified model across various speech datasets is outlined in Table 2a, where area under the receiver operating characteristic (AUROC) is used as the metric. For the DIHARD3 [23] dataset, we excluded Conversational Telephonic Speech (CTS) and computed a macro-average across ten different domains, as the CTS domain is derived from the Fisher dataset, which possesses significantly loose timestamps. Through these modifications, we achieved an overall high performance with our model on the four datasets, especially with the application of noise augmentation and gain perturbation.

Several significant observations arise from our experiments with VAD models using the simulated dataset. Firstly, loose timestamps, which encapsulate non-speech signals at the start and end of each segment, can markedly degrade the performance of VAD. This issue is exacerbated by data augmentation, as the model is then trained with the added noise at the boundaries of each segment. Secondly, gain perturbation is a necessary consideration as the VAD model frequently overlooks low-volume speech signals. To mitigate this, the model should be trained with substantial variation in gain during the creation of audio mixtures. Lastly, the addition of overlapping speech is also essential for enhancing performance, as overlapping speech can lead to an increase in missed detections.

[3] https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/vad_multilingual_frame_marblenet

Table 2: *Evaluation of models on different parameters*

(a) *AUROC for VAD task*

Training Data / Dataset Split	DH3 dev	VoxConv dev	AMI dev	CH109 -
D1,μ_s=0.5	87.71	96.15	95.7	88.07
D1,μ_s=0.3	89.83	96.19	94.69	91.04
+ Gain. Aug.	93.7	96.02	**96.55**	88.73
+ D2 + Noise. Aug.	**93.96**	**97.42**	96.04	**92.43**

(b) *DER(%) on Diarization Datasets*

Training Data / Dataset Split	DH3 eval	VoxConv test	AMI eval	CH-109 -
LibriVox-3Kh μ_o=0.07	14.49	6.01	15.96	**9.94**
LibriVox-3Kh μ_o=0.15	**14.38**	**5.72**	**15.89**	10.03

(c) *DER(%) on CHiME7 Datasets*

Training Data / Dataset Split	Chime6 dev	Dipco dev	Mixer6 dev
LibriVox-3Kh μ_o=0.07	45.01	32.50	17.35
LibriVox-3Kh μ_o=0.15	**44.37**	**31.07**	**17.13**

3.3. Speaker Diarization Model

As in the previous section, we train a modified speaker diarization model, based on [24], alongside the speaker embedding model from [25]. The experiment utilizes 1k hours of LibriSpeech and 2k hours of VoxCeleb 1 and 2 [26]. We use the same type of time stamps and random word-level alignment as in the Fisher dataset for Voice Activity Detection (VAD). For diarization evaluation, we use the VAD model from Table 2a. Diarization error rate (DER) is calculated using a 0.25 sec collar, with overlap considered. DER is assessed on DIHARD3, VoxConverse-3 [27], AMI eval(test)-sets, and 2-speaker CHAES subset, CH109. Tables 2b and 2c show performance variations with different synthetic dataset settings.

4. Conclusions

In this paper, we introduce a property-aware data simulator capable of reflecting statistics provided by the user or extracted from real-world data. The proposed data simulator utilizes an online sampling technique, allowing the system to generate a predetermined quantity of silence and overlap speech while adhering to the given probability distributions. Consequently, the generated dataset can be leveraged to train VAD models and speaker diarization models, providing highly accurate ground-truth timestamps, which is a critical element for both speech activity detection and speaker diarization. Potential future research could involve adapting this system for online generation, whereby users could supply a source dataset and generate the training dataset on-the-fly. We anticipate that the proposed data simulator will be adopted by the speech signal processing community for training neural models related to speech signals, which necessitate accurate ground truth timestamps and highly customizable speech training data.

5. References

[1] T. J. Park, N. Kanda, D. Dimitriadis, K. J. Han, S. Watanabe, and S. Narayanan, "A review of speaker diarization: Recent advances with deep learning," *Computer Speech & Language*, vol. 72, p. 101317, 2022.

[2] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019.

[3] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Sixteenth annual conference of the international speech communication association*, 2015.

[4] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. CONF. IEEE Signal Processing Society, 2011.

[5] Y. Fujita, N. Kanda, S. Horiguchi, Y. Xue, K. Nagamatsu, and S. 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.

[6] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2016, pp. 31–35.

[7] S. Horiguchi, Y. Fujita, S. Watanabe, Y. Xue, and K. Nagamatsu, "End-to-end speaker diarization for an unknown number of speakers with encoder-decoder based attractors," *arXiv preprint arXiv:2005.09921*, 2020.

[8] F. Landini, A. Lozano-Diez, M. Diez, and L. Burget, "From simulated mixtures to simulated conversations as training data for end-to-end neural diarization," *Interspeech*, 2022.

[9] F. Landini, M. Diez, A. Lozano-Diez, and L. Burget, "Multi-speaker and wide-band simulated conversations as training data for end-to-end neural diarization," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[10] B. D. Fielitz and B. L. Myers, "Concepts, theory, and techniques: Estimation of parameters in the beta distribution," *Decision Sciences*, vol. 6, no. 1, pp. 1–13, 1975.

[11] M. Heldner and J. Edlund, "Pauses, gaps and overlaps in conversations," *Journal of Phonetics*, vol. 38, no. 4, pp. 555–568, 2010.

[12] H. Jin and H. Liu, "How will text size influence the length of its linguistic constituents?" *Poznan Studies in Contemporary Linguistics*, vol. 53, no. 2, pp. 197–225, 2017.

[13] W. Kraaij, T. Hain, M. Lincoln, and W. Post. (2005) The ami meeting corpus. Accessed: 2023-06-16. [Online]. Available: http://https://groups.inf.ed.ac.uk/ami/corpus

[14] A. Canavan, D. Graff, and G. Zipperlen, "Callhome american english speech corpus ldc97s42," DVD, 1997, available from Linguistic Data Consortium, University of Pennsylvania.

[15] F. Jia, S. Majumdar, and B. Ginsburg, "Marblenet: Deep 1d time-channel separable convolutional neural network for voice activity detection," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6818–6822.

[16] C. Cieri, D. Miller, and K. Walker, "The fisher corpus: A resource for the next generations of speech-to-text," in *LREC*, vol. 4, 2004, pp. 69–71.

[17] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[18] C. Jemine. (2023) Librispeech alignments. [Online]. Available: https://github.com/CorentinJ/librispeech-alignments

[19] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," *arXiv preprint arXiv:1912.06670*, 2019.

[20] Sovaai, "Sova dataset," 2023. [Online]. Available: https://github.com/sovaai/sova-dataset

[21] J. Du, X. Na, X. Liu, and H. Bu, "Aishell-2: Transforming mandarin asr research into industrial scale," *arXiv preprint arXiv:1808.10583*, 2018.

[22] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, "Mls: A large-scale multilingual dataset for speech research," *arXiv preprint arXiv:2012.03411*, 2020.

[23] N. Ryant, P. Singh, V. Krishnamohan, R. Varma, K. Church, C. Cieri, J. Du, S. Ganapathy, and M. Liberman, "The third dihard diarization challenge," *arXiv preprint arXiv:2012.01477*, 2020.

[24] T. J. Park, N. R. Koluguri, J. Balam, and B. Ginsburg, "Multi-scale speaker diarization with dynamic scale weighting," *Interspeech*, 2022.

[25] N. R. Koluguri, T. Park, and B. Ginsburg, "Titanet: Neural model for speaker representation with 1d depth-wise separable convolutions and global context," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 8102–8106.

[26] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," in *Interspeech*. International Speech Communication Association, 2017.

[27] J. Huh, A. Brown, J.-w. Jung, J. S. Chung, A. Nagrani, D. Garcia-Romero, and A. Zisserman, "Voxsrc 2022: The fourth voxceleb speaker recognition challenge," *arXiv preprint arXiv:2302.10248*, 2023.