



HiFiTTS-2: A Large-Scale High Bandwidth Speech Dataset

Ryan Langman¹, Xuesong Yang¹, Paarth Neekhara¹, Shehzeen Hussain¹, Edresson Casanova¹,
Evelina Bakhturina¹, Jason Li¹

¹NVIDIA, USA

{rlangman, xueyang, pneekhara, shehzeen, ecasanova, ebakhturina, jasoli}@nvidia.com

Abstract

This paper introduces HiFiTTS-2, a large-scale speech dataset designed for high-bandwidth speech synthesis. The dataset is derived from LibriVox audiobooks, and contains approximately 36.7k hours of English speech for 22.05 kHz training, and 31.7k hours for 44.1 kHz training. We present our data processing pipeline, including bandwidth estimation, segmentation, text preprocessing, and multi-speaker detection. The dataset is accompanied by detailed utterance and audiobook metadata generated by our pipeline, enabling researchers to apply data quality filters to adapt the dataset to various use cases. Experimental results demonstrate that our data pipeline and resulting dataset can facilitate the training of high-quality, zero-shot text-to-speech (TTS) models at high bandwidths.

Index Terms: text-to-speech, speech synthesis, multi-speaker dataset

1. Introduction

Text-to-speech (TTS) and speech synthesis technology have advanced rapidly in recent years. Initially, TTS research focused on creating a small number of high-quality voices for applications like voice assistants. Datasets developed for this purpose, such as LJSpeech [1] and HiFiTTS [2], typically contained large amounts of audio from one or a few speakers, emphasizing a need for clean or studio-quality audio at high frequency resolutions like 22.05 kHz or 44.1 kHz.

More recently, research has shifted towards zero-shot TTS, which aims to replicate the voice of a speaker not seen during training, using only a few seconds of reference audio. A common approach for zero-shot TTS is training an autoregressive language model to predict audio tokens produced by a neural audio codec [3, 4, 5, 6, 7, 8, 9]. These approaches rely on large amounts of transcribed training data, which is challenging to obtain in the speech domain. Consequently, most recent research uses large volumes of low-quality 16 kHz audio [4, 9, 7]. While these models achieve high performance for zero-shot TTS, they synthesize speech with bandwidth and subsequent quality too low for many real-world applications.

Modern TTS models have different data requirements compared to earlier models. While previous models often required clean or studio-quality audio [10, 11] due to their inability to accurately model noise or acoustics, current models can learn these speech characteristics through in-context learning [3]. For this reason, our data processing does not focus on signal-to-noise ratio (SNR) metrics, which were emphasized in earlier works and datasets like HiFiTTS.

Several challenges arise when using speech data from diverse real-world sources. There are significant legal complexities surrounding data use for applications like speech synthe-

sis and voice-cloning. For instance, Emilia [12], a large-scale speech dataset, uses a non-commercial license due to its underlying sources having varied licenses and copyright. In contrast, our dataset is suitable for commercial use, as it derives from LibriVox [13] audiobooks in the public domain. We remove data from LibriVox speakers who have explicitly requested their recordings not be used for machine learning applications.

Another challenge comes from the common practice of storing audio in a standardized format regardless of how it was recorded. This often results in audio being upsampled to a higher resolution than its original recording. Many publicly available datasets with high sampling rates, such as CommonVoice [14], Emilia [12], and DNS [15], contain such mixed-bandwidth audio. LibriVox audiobooks are stored at 24 kHz and 48 kHz, but datasets derived from them are often downsampled to 16 kHz, as seen in Librispeech [16], MLS [14], and LibriLight [17]. Mixed-bandwidth audio can be problematic for training speech models. For example, studies have shown that training audio codecs on mixed-bandwidth data results in attenuation of all audio information above a certain frequency threshold [18, 19]. Training with mixed-bandwidth data benefits from bandwidth labels, which can be used to filter data [18], be provided as input to models [20], or used for bandwidth extension [21].

In this paper, we present HiFiTTS-2, a large-scale audiobook dataset designed for speech synthesis. Our contributions are as follows:

- We present a data processing pipeline consisting of bandwidth estimation, segmentation, text processing and validation, and multi-speaker detection.
- We create and release a dataset¹ with two subsets by applying this pipeline to LibriVox audiobooks: a 22.05 kHz subset with 36.7k hours of speech from 5k speakers, and a 44.1 kHz subset with 31.7k hours of speech from 4.6k speakers.
- We provide precomputed metadata at the utterance and audiobook chapter level, which users can use to apply quality control filters optimized for their individual use cases.
- We demonstrate the effectiveness of our data pipeline by conducting experiments with a state-of-the-art TTS model.

2. Data Processing Pipeline

We use the English subset of MLS [14] as the input to our pipeline. MLS is a multi-lingual dataset derived from LibriVox audiobooks, primarily designed for ASR model training. The English subset of MLS contains approximately 44.7k hours of speech from 5,574 speakers. However, several significant challenges arise when using the original dataset for training TTS

¹<https://huggingface.co/datasets/nvidia/hifitts-2>

Table 1: Text preprocessing examples applied to original MLS input. The process restores punctuation and capitalization, normalizes text, and replaces formatting elements such as ‘nbsp’ and ‘p p’.

| Input Text | Preprocessed Text |
|--|---|
| beautifully-shaped and coloured glass and saltcellars tankards c of gold and silver | beautifully shaped and coloured glass , and saltcellars , tankards , et cetera of gold and silver . |
| at that moment of supreme anxiety nbsp it is my purpose nbsp mr allen read | at that moment of supreme anxiety . “It is my purpose ,” mister Allen read |
| in his calculations p p rather would i believe that i have been mistaken in the affection which i feel for him said mrs evangelina | in his calculations . “Rather would I believe that I have been mistaken in the affection which I feel for him ,” said misses Evangelina |

models:

- Text transcriptions lack punctuation and capitalization, which plays crucial roles in conveying prosodic features in TTS.
- All audio data is stored at a 16 kHz sampling rate, regardless of their original bandwidth.
- There are no utterances shorter than 10 seconds, with all utterances ranging from 10 to 20 seconds in length.
- Text transcriptions are taken directly from the audiobook text, which may differ slightly from the speaker’s narration.
- Speaker labels may be unreliable, as some audiobooks are narrated by multiple speakers.

In the following sections, we describe our processing pipeline to address these issues. Processing steps are carried out in the order listed.

2.1. Text Preprocessing

We start with the utterances and their transcripts from the English subset of MLS. To recover punctuation and capitalization (PC) for the transcripts, we take two approaches:

1. We download the original audiobook text for each chapter. After removing PC from the audiobook text, we check whether each MLS transcript matches a substring in the text. If a match is found, we replace the original MLS transcript with the corresponding audiobook text. This method successfully recovers PC for approximately 87% of the original MLS transcripts.
2. For the remaining 13% of transcripts, we predict PC using NeMo DistilBERT [22].

In our dataset we provide metadata specifying which utterance transcriptions are from the audiobook and which are from MLS with predicted PC. After restoring PC to the text, we attempt to remove various formatting information common in the downloaded audiobook text, such as HTML tags. Lastly, we normalize the text using NeMo text normalization [23]. Table 1 shows a few examples of text preprocessing.

2.2. Audio Processing

To process the audio for the dataset, we first download all of the original 48 kHz audiobook files using the LibriVox API. We downsample the audio to 44.1 kHz and convert files from MP3 to FLAC format. We apply energy-based silence trimming with a threshold of 50 dB, leaving at most 0.5 seconds of silence at the start and end of each utterance.

2.3. Bandwidth Estimation

We apply the bandwidth estimation approach from the Hi-FiTTS [2] to the first 30 seconds of each audiobook. The bandwidth f_{\max} is estimated by using the mean of the power spectrum to find the highest frequency that has at least -50 dB level relative to the peak value of the spectrum, namely,

$$f_{\max} = \max \left\{ f \in [0, f_{\text{Nyquist}}] \mid 10 \log_{10} \left(\frac{P(f)}{P_{\text{peak}}} \right) \geq -50 \text{ dB} \right\}$$

where $P(f)$ is the power spectral density and P_{peak} the maximum spectral power. Utterances will have an estimated bandwidth that is at most its Nyquist frequency. So audio recorded at 16 kHz sampling rate will have estimated bandwidth less than or equal to 8 kHz and audio recorded at 24 kHz will have estimated bandwidth less than or equal to 12 kHz.

For the 22.05 kHz subset of our dataset, we filter for utterances with an estimated bandwidth of at least 11 kHz. This produces approximately 36.7k hours of full-bandwidth 22.05 kHz speech.

Processing 44.1 kHz audio introduces additional complexity. Most speech recorded at a sampling rate of 24 kHz or lower is close to full-bandwidth. However, speech recorded at higher sampling rates like 48 kHz often has little spectral information in the highest frequency bands, resulting in an estimated bandwidth that is significantly lower than its Nyquist frequency. This results in most data for 44.1 kHz or 48 kHz training being effectively mixed-bandwidth. Training models on mixed-bandwidth data is challenging, and research on it is relatively uncommon due to most large public datasets being 16 kHz sampling rate.

To address this, we filter 44.1 kHz audio using a bandwidth filter of 13 kHz, similar to Hi-FiTTS [2]. This removes all audio recorded at a sampling rate of 24 kHz or lower. We assume that recording at sampling rates between 24 kHz and 44.1 kHz is very rare, even for data in-the-wild. This process yields approximately 31.7k hours of 44.1 kHz speech with bandwidth ranging from approximately 13 kHz to 22 kHz. We release this subset of the data, along with metadata containing the estimated bandwidth, to assist with future research on high bandwidth and mixed bandwidth modeling. The resulting bandwidth distribution is shown in Figure 1.

2.4. Segmentation

Utterances in the MLS data range from 10 to 20 seconds in length. However, for TTS training a more balanced distribution of durations is desirable to prevent biases in model performance. To achieve this, we employ the NeMo Forced Aligner [24] with Parakeet-TDT-CTC [25] to generate align-

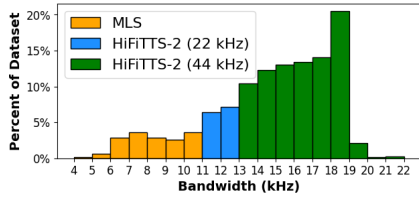


Figure 1: *Distribution of estimated bandwidth across utterances. Orange shows low-bandwidth utterances in MLS that are excluded from HiFiTTS-2. Green denotes high-bandwidth utterances present in all datasets. Blue denotes utterances included in the 22 kHz subset but absent from the 44 kHz subset.*

ment information. Using these alignments, we identify utterances containing a period followed by a pause of at least 0.08 seconds, ignoring periods within abbreviations. We then split such utterances at the midpoint of the pause, similar to the segmentation methodology from MLS [14]. If multiple qualifying pauses exist within an utterance, we select the longest one for splitting. In cases where multiple pauses have the same duration, we choose one at random. This process results in a duration distribution resembling a bell curve, as shown in Figure 2.

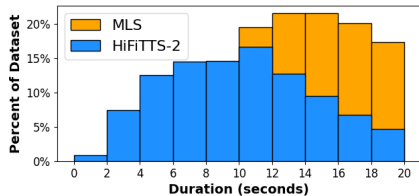


Figure 2: *Distribution of utterance durations. Orange denotes the original distribution before segmentation.*

2.5. Text Validation

We validate the correctness of the final text using ASR. For all segmented utterances, we compute WER (word error rate) and CER (character error rate) using predicted transcriptions from Parakeet-TDT-CTC [25]. We filter out utterances with a CER of 100% or greater, which correspond to utterances with incorrect transcripts or utterances containing only silence. Applying any higher threshold creates a trade-off between the volume and intelligibility of the training data. We provide the WER and CER as metadata in the dataset, allowing users to filter using a threshold appropriate for their use case. The utterance distribution of WER and CER is shown in Figure 3.

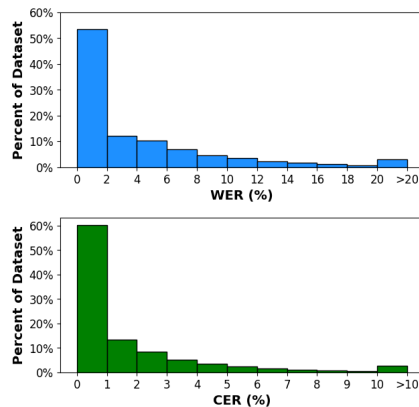


Figure 3: *Distribution of WER and CER in HiFiTTS-2.*

2.6. Speaker Counts

Some LibriVox audiobooks are narrated by multiple speakers, leading to utterances that are not always ideal for modeling purposes. In order to identify and quantify the number of speakers present in each utterance, we employ Softformer [26], a recent end-to-end neural model for speaker diarization. This model efficiently detects instances of multi-speaker speech, allowing us to filter and annotate such segments accordingly. Through this process, we identify approximately 104.4 hours of speech in the dataset as containing multiple speakers. We tag the number of speakers for each utterance as metadata in our dataset.

3. Dataset Statistics

HiFiTTS-2 comprises utterances from LibriVox audiobooks, with each utterance capped at 20 seconds in length. The dataset contains two subsets:

1. A full-bandwidth 22.05 kHz subset containing 36.7k hours of speech with 13.1M utterances from 5,013 speakers;
2. A 44.1 kHz subset with bandwidths above 13 kHz containing 31.7k hours of speech with 11.3M utterances from 4,631 speakers.

These subsets are overlapping, with the 44.1 kHz being a strict subset of the 22.05 kHz.

Table 2 provides a comparison between HiFiTTS-2 and other datasets commonly used for speech synthesis, highlighting its significant scale advantage.

We provide two sets of dev and test splits: one for speakers seen in the training data and the other one for unseen speakers not present in the training data. To ensure consistency and maximize data quality, we use the same utterances for both the 22.05 kHz and 44.1 kHz subsets, selecting only utterances with at least 13 kHz bandwidth, zero WERs, and a single speaker.

For unseen speakers, we use utterances produced by passing the MLS dev and test partitions through our data processing pipeline. This creates an unseen speaker dev set with 1098 utterances from 29 speakers and a test set with 968 utterances from 27 speakers.

For seen speakers, we create dev and test partitions each consisting of 1,000 utterances. These utterances were selected by sampling 50 speakers with 15 to 60 minutes of training audio, and sampling 20 utterances from each speaker such that utterances are uniformly distributed across gender, duration, and bandwidth.

These carefully curated splits facilitate the robust evaluation of model performance, important for assessing the generalization of speech synthesis systems.

Table 2: *Statistics for HiFiTTS-2 compared to other datasets.*

| Dataset | Sampling Rate | Hours | Speakers |
|--------------------|---------------|-------|----------|
| VCTK | 48 kHz | 44 | 110 |
| HiFiTTS | 44.1 kHz | 292 | 10 |
| LibriTTS | 24 kHz | 586 | 2,456 |
| MLS (English) | 16 kHz | 44.7k | 5,574 |
| HiFiTTS-2 (22 kHz) | 22.05 kHz | 36.7k | 5,013 |
| HiFiTTS-2 (44 kHz) | 44.1 kHz | 31.7k | 4,631 |

4. Experiments

To demonstrate the benefit of using HiFiTTS-2 for speech synthesis, we employ Koel-TTS [6], a state-of-the-art encoder-decoder architecture. We compare performance at 22.05 kHz against existing LibriVox datasets: LibriTTS and HiFiTTS.

Table 3: *Koel-TTS results. Note that the model trained exclusively on HiFiTTS has no seen speakers in the test set.*

| Train Datasets | Dur (hrs) | #Spks | Seen Speakers | | | | Unseen Speakers | | | |
|------------------------------|-----------|-------|------------------|------------------|--------------------|--------------------|------------------|------------------|--------------------|--------------------|
| | | | CER (%)↓ | WER (%)↓ | SSIM ↑ | SQUIM-MOS ↑ | CER (%)↓ | WER (%)↓ | SSIM ↑ | SQUIM-MOS ↑ |
| <i>Ground Truth (oracle)</i> | | | <i>0.51±0.00</i> | <i>1.42±0.00</i> | <i>0.763±0.000</i> | <i>4.616±0.030</i> | <i>0.80±0.00</i> | <i>1.83±0.00</i> | <i>0.771±0.000</i> | <i>4.588±0.020</i> |
| HiFiTTS | 283 | 10 | – | – | – | – | 6.97±0.37 | 10.38±0.36 | 0.059±0.004 | 4.380±0.005 |
| LibriTTS | 539 | 2,259 | 0.92±0.22 | 2.13±0.30 | 0.550±0.002 | 4.390±0.005 | 1.44±0.27 | 2.48±0.26 | 0.494±0.003 | 4.383±0.004 |
| HiFiTTS, LibriTTS | 822 | 2,265 | 0.77±0.13 | 1.72±0.17 | 0.723±0.002 | 4.461±0.013 | 0.80±0.07 | 1.67±0.11 | 0.588±0.002 | 4.414±0.011 |
| HiFiTTS-2 | 30,400 | 4,940 | 0.47±0.11 | 1.21±0.12 | 0.714±0.002 | 4.388±0.004 | 0.57±0.11 | 1.42±0.13 | 0.731±0.001 | 4.387±0.005 |
| HiFiTTS, LibriTTS, HiFiTTS-2 | 31,222 | 5,657 | 0.62±0.15 | 1.43±0.17 | 0.739±0.002 | 4.385±0.007 | 0.57±0.04 | 1.39±0.08 | 0.739±0.002 | 4.382±0.006 |

4.1. Koel-TTS Model

Koel-TTS [6] consists of an autoregressive (AR) transformer decoder conditioned on text encodings from a non-autoregressive (NAR) transformer encoder, using cross-attention layers. The model has approximately 378M trainable parameters. The AR transformer predicts audio tokens frame by frame, generating all codebooks in parallel at each time step, conditioned on the input text and previous audio tokens. As shown in Figure 4, the context audio tokens are directly provided as input to the AR decoder by prepending them to the target audio tokens. A single unified transformer decoder processes both the context and target audio tokens, leveraging a shared representation for conditioning and prediction. We adapted classifier-free guidance (CFG) [6] to train models by dropping out the text and audio conditionals with a probability of 10%. During inference, we applied a CFG scale of 2.5 to guide the AR token prediction for more precise control over the generated speech.

Our models are trained on 32 NVIDIA DGX H100 GPUs using a global batch size of 512. We use an Adam optimizer with an initial learning rate $1e-5$ for small-scale training with LibriTTS and HiFiTTS, and $2e-4$ for large-scale training with HiFiTTS-2 and their combinations. No weight decays are applied. The learning rate is annealed every 1,000 training steps with an exponential decay factor of 0.998.

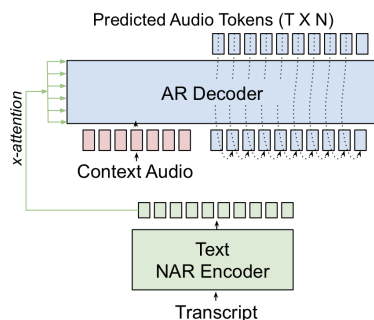


Figure 4: *Koel-TTS Model Architecture*

4.2. Evaluation Metrics and Data

We evaluated the synthesized speech on intelligibility, speaker similarity, and audio quality. Intelligibility is measured with character error rate (CER) and word error rate (WER) computed by a state-of-the-art ASR Model Parakeet-TDT [25]. Speaker similarity is measured with cosine similarity between synthesized and context audio embeddings extracted by a speaker recognition model Titanet-Small [27]. Audio quality is measured with a reference-free MOS estimator SQUIM-MOS [28]. We use multinomial top-k sampling with $k=80$ and $temperature=0.6$ during inference, and report mean metrics with 95% confidence intervals after running 10 times for

each experiment.

For **training data**, we train one model for each individual dataset: LibriTTS, HiFiTTS, and HiFiTTS-2. Both *clean* and *other* subsets are chosen for LibriTTS and HiFiTTS. We also experiment with combining datasets. For each dataset, we manually create triplets of (5-second context audio, transcript, target audio), where context and target audio are distinct utterances from the same speaker. For HiFiTTS-2, we also discarded the triplets where target audio CER is greater than 3% or speaker similarity between context and target audio is less than 0.6. This filtering results in 30,400 hours of training data.

For **test data**: We evaluate models on subsets of LibriTTS for both seen and unseen speakers. We withhold 200 utterances from 170 speakers in *train-clean-360* as seen speakers, and 180 utterances from 36 speakers in *test-clean* as unseen speakers. We use 5 distinct context and target audio pairs from each speaker.

4.3. Results and Analysis

Table 3 illustrates the comparison between Koel-TTS models trained on individual datasets and their combination. We observed that the model trained on LibriTTS serves as a strong baseline, performing well on metrics for both seen and unseen speakers. As expected, the model trained solely on HiFiTTS underperforms on unseen speakers due to its limited speaker diversity covering only 10 speakers, as shown in Table 2. Combining LibriTTS and HiFiTTS improves performance on both seen and unseen speakers, achieving performance on SQUIM-MOS slightly better than even the large-scale trainings. However, the most substantial improvements come from training with HiFiTTS-2, which significantly enhances performance, especially in speaker similarity for unseen speakers. This highlights the benefits of our proposed large-scale dataset. Furthermore, training with all three available datasets led to additional improvements, particularly for unseen speakers, emphasizing the critical role of diverse, large-scale data for zero-shot TTS.

5. Conclusion

In this paper we introduce HiFiTTS-2, a large-scale English dataset designed for modeling high-bandwidth speech. Derived from LibriVox audiobooks originally downloaded at 48 kHz, the dataset is processed through our custom pipeline to assess and ensure the quality of both audio and transcripts. Our experiments demonstrate the effectiveness of HiFiTTS-2 and our data processing methods for TTS applications, and we believe its utility can extend to other domains such as bandwidth extension. In future, we aim to expand our work to incorporate data from diverse sources and languages, and to explore applications beyond TTS that require high-bandwidth audio processing.

6. References

- [1] K. Ito and L. Johnson. (2017) The LJ speech dataset. <https://keithito.com/LJ-Speech-Dataset>.
- [2] E. Bakhturina, V. Lavrukhin, B. Ginsburg, and Y. Zhang, “Hi-Fi multi-speaker English TTS dataset,” in *Interspeech*, 2021, pp. 2776–2780.
- [3] S. Chen, C. Wang, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, L. He, S. Zhao, and F. Wei, “Neural codec language models are zero-shot text to speech synthesizers,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 705–718, 2025.
- [4] X. Zhang, D. Zhang, S. Li, Y. Zhou, and X. Qiu, “SpeechTokenizer: Unified speech tokenizer for speech language models,” in *International Conference on Learning Representations (ICLR)*, 2024.
- [5] P. Neekhara, S. Hussain, S. Ghosh, J. Li, and B. Ginsburg, “Improving robustness of LLM-based speech synthesis by learning monotonic alignment,” in *Interspeech*, 2024, pp. 3425–3429.
- [6] S. Hussain, P. Neekhara, X. Yang, E. Casanova, S. Ghosh, M. T. Desta, R. Fejgin, R. Valle, and J. Li, “Koel-TTS: Enhancing LLM based speech generation with preference alignment and classifier free guidance,” *arXiv:2502.05236*, 2025.
- [7] S. Chen, S. Liu, L. Zhou, Y. Liu, X. Tan, J. Li, S. Zhao, Y. Qian, and F. Wei, “VALL-E 2: Neural codec language models are human parity zero-shot text to speech synthesizers,” *arXiv:2406.05370*, 2024.
- [8] E. Casanova, R. Langman, P. Neekhara, S. Hussain, J. Li, S. Ghosh, A. Jukić, and S.-g. Lee, “Low frame-rate speech codec: a codec designed for fast high-quality speech LLM training and inference,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2025.
- [9] P. Peng, P.-Y. B. Huang, S.-W. Li, A. Mohamed, and D. Harwath, “VoiceCraft: Zero-shot speech editing and text-to-speech in the wild,” in *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.
- [10] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, “Natural TTS synthesis by conditioning Wavenet on MEL spectrogram predictions,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.
- [11] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, “FastSpeech: fast, robust and controllable text to speech,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.
- [12] H. He, Z. Shang, C. Wang, X. Li, Y. Gu, H. Hua, L. Liu, C. Yang, J. Li, P. Shi *et al.*, “Emilia: An extensive, multilingual, and diverse speech dataset for large-scale speech generation,” in *Spoken Language Technology Workshop (SLT)*. IEEE, 2024, pp. 885–890.
- [13] Librivox - free public domain audiobooks. <https://librivox.org>.
- [14] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, “MLS: A large-scale multilingual dataset for speech research,” in *Interspeech*, 2020, pp. 2757–2761.
- [15] H. Dubey, A. Aazami, V. Gopal, B. Naderi, S. Braun, R. Cutler, A. Ju, M. Zohourian, M. Tang, M. Golestaneh *et al.*, “ICASSP 2023 deep noise suppression challenge,” in *Open Journal of Signal Processing*. IEEE, 2024, pp. 725–737.
- [16] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: an ASR corpus based on public domain audio books,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.
- [17] J. Kahn, M. Riviere, W. Zheng, E. Kharitonov, Q. Xu, P.-E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen *et al.*, “Libri-Light: A benchmark for ASR with limited or no supervision,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7669–7673.
- [18] R. Kumar, P. Seetharaman, A. Luebs, I. Kumar, and K. Kumar, “High-fidelity audio compression with improved RVQGAN,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2024.
- [19] K. C. Puvvada, N. R. Koluguri, K. Dhawan, J. Balam, and B. Ginsburg, “Discrete audio representation as an alternative to Mel-spectrograms for speaker and speech recognition,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 12 111–12 115.
- [20] G. Mantena, O. Kalinli, O. Abdel-Hamid, and D. McAllister, “Bandwidth embeddings for mixed-bandwidth speech recognition,” in *Interspeech*, 2019, pp. 3203–3207.
- [21] J. Su, Y. Wang, A. Finkelstein, and Z. Jin, “Bandwidth extension is all you need,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 696–700.
- [22] NVIDIA. (2023) Punctuation En Distilbert. https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/stt_en_fastconformer_transducer_xlarge.
- [23] Y. Zhang, E. Bakhturina, K. Gorman, and B. Ginsburg, “NeMo (inverse) text normalization: From development to production,” in *Interspeech*, 2021, pp. 4468–4472.
- [24] E. Rastorgueva, V. Lavrukhin, and B. Ginsburg, “NeMo forced aligner and its application to word alignment for subtitle generation,” in *Interspeech*, 2023, pp. 5257–5258.
- [25] H. Xu, F. Jia, S. Majumdar, H. Huang, S. Watanabe, and B. Ginsburg, “Efficient sequence transduction by jointly predicting tokens and durations,” in *International Conference on Machine Learning (ICML)*. PMLR, 2023. [Online]. Available: <https://huggingface.co/nvidia/parakeet-tdt-1.1b>
- [26] T. Park, I. Medennikov, K. Dhawan, W. Wang, H. Huang, N. R. Koluguri, K. C. Puvvada, J. Balam, and B. Ginsburg, “Sortformer: Seamless integration of speaker diarization and ASR by bridging timestamps and tokens,” *arXiv:2409.06656*, 2024. [Online]. Available: https://huggingface.co/nvidia/diar_sortformer_4spk-v1
- [27] N. R. Koluguri, T. Park, and B. Ginsburg, “TitaNet: Neural model for speaker representation with 1D depth-wise separable convolutions and global context,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 8102–8106. [Online]. Available: https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/titanet_small
- [28] A. Kumar, K. Tan, Z. Ni, P. Manocha, X. Zhang, E. Henderson, and B. Xu, “Torchaudio-Squim: Reference-less speech quality and intelligibility measures in Torchaudio,” in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.