Unsupervised Text-to-Speech Synthesis by Unsupervised Automatic Speech Recognition

Junrui Ni[1], Liming Wang[1*], Heting Gao[1*], Kaizhi Qian[2], Yang Zhang[2], Shiyu Chang[3], Mark Hasegawa-Johnson[1]*

[1]University of Illinois at Urbana-Champaign
[2]MIT-IBM Watson AI Lab [3]University of California, Santa Barbara
{junruin2,lwang114,hgao17,jhasegaw}@illinois.edu, {kqian, yang.zhang2}@ibm.com,
chang87@ucsb.edu

Abstract

An unsupervised text-to-speech synthesis (TTS) system learns to generate speech waveforms corresponding to any written sentence in a language by observing: 1) a collection of untranscribed speech waveforms in that language; 2) a collection of texts written in that language without access to any transcribed speech. Developing such a system can significantly improve the availability of speech technology to languages without a large amount of parallel speech and text data. This paper proposes an unsupervised TTS system based on an alignment module that outputs pseudo-text and another synthesis module that uses pseudo-text for training and real text for inference. Our unsupervised system can achieve comparable performance to the supervised system in seven languages with about 10-20 hours of speech each. A careful study on the effect of text units and vocoders has also been conducted to better understand what factors may affect unsupervised TTS performance. The samples generated by our models can be found at https://cactuswiththoughts. github.io/UnsupTTS-Demo, and our code can be found at https://github.com/lwang114/UnsupTTS.

Index Terms: speech synthesis, speech recognition, unsupervised learning

1. Introduction

Text-to-speech (TTS) synthesis is an essential component of a spoken dialogue system. While capable of generating high-fidelity, human-like speech for languages such as English and Mandarin, the existing state-of-the-art TTS systems such as Tacotron 1&2 [1, 2], Deep Voice 3 [3], FastSpeech [4] and Transformer TTS [5] are trained with a large amount of parallel speech and textual data. The reliance on a large amount of transcribed speech makes such systems impractical for the majority of the languages in the world. Training a supervised text-to-speech (TTS) system requires dozens of hours of single-speaker high-quality recordings [6], but collecting a large amount of single-speaker, clean, and transcribed speech corpus can be quite time-consuming and expensive [7]. A potential way to relax such a requirement is to use *non-parallel* untranscribed speech and text corpora in the same language. Such corpora are much easier to obtain in practice since no human annotators are required in the data collection process, thanks to the abundance of text data on the Internet. Learning to perform TTS using non-parallel speech and text, or *unsupervised TTS*, poses unique challenges: first, standard supervised training criteria such as autoregressive mean-squared error are no longer

applicable; further, to learn a latent alignment between the spoken frames and text units, the model now needs to search over every utterance and every transcript in the entire corpus instead of limiting the search space within a single utterance-transcript pair.

This paper proposes the first model for solving the unsupervised TTS problem. We decompose training the model into two tasks, learning an alignment module that assigns a single pseudo-transcript to each utterance and learning a synthesis module that learns from pseudo-text and utterance pairs. The alignment module is motivated by the best publicly available unsupervised ASR system, wav2vec-U [8], and can generalize seamlessly to future updates of the wav2vec-U model. We conduct our unsupervised TTS experiments on seven languages. We further provide an in-depth analysis of the effect of several components on unsupervised TTS performance, including the grapheme-to-phoneme (G2P) converter and the vocoder.

2. Related works

Several recent works have attempted to develop TTS systems for low-resource scenarios. One direction of research is to replace ground truth phoneme or grapheme labels required for supervised TTS with other units obtained with less or no supervision, such as articulatory features [9], or acoustic units discovered by self-supervised speech representation models such as vector-quantized variational auto-encoder (VQ-VAE) [10, 11] and HuBERT [12, 13, 14]. The unsupervised, textless approach can be applied to any language, including those without any written form. However, the performance of such a system is limited by the quality of the acoustic units used, which can be quite noisy due to the difficulty of acoustic unit discovery. To address this limitation, [15, 16, 17, 18] have studied the use of other sensory modalities such as images in place of textual transcripts as a weaker form of supervision for conditional generation of speech, or "TTS without T", using various attention mechanisms over the visual features. Another approach to address this issue is to allow a small amount of transcribed speech and train the TTS in a semi-supervised fashion [19, 6]. Specifically, [19] leveraged unpaired speech and text data by constructing pseudo-corpora via dual transformation between ASR and TTS systems with on-the-fly refinement followed by knowledge distillation, while LRSpeech [6] trained an ASR and a TTS system that used only several minutes of paired single-speaker, high-quality speech for TTS, and several hours of low-quality, multi-speaker data for ASR.

Our approach is motivated by the most recently published unsupervised automatic speech recognition (ASR) system [8],
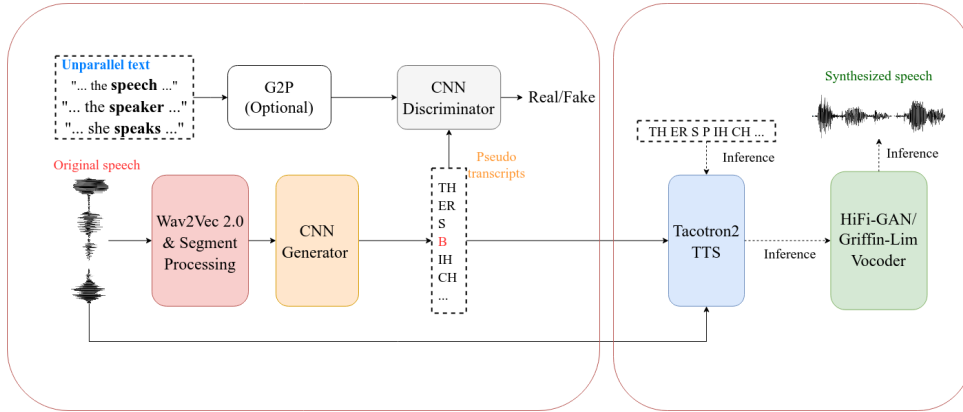

Figure 1: *Network architecture for unsupervised speech synthesis, splitted into an alignment module (left) and a synthesis module (right)*

which learns to recognize phones by leveraging pre-trained speech representations and an unpaired text corpus. Other works on unsupervised ASR typically try to match the empirical prior and posterior distributions of phonemes either using cross-entropy [20] or adversarial loss [21]. Using powerful self-supervised, pre-trained acoustic features such as wav2vec 2.0 [22] and a generative adversarial network (GAN)-based system, the adversarial approach with self-training achieves comparable performance to its supervised counterpart on large-scale speech datasets for multiple languages [8].

3. Proposed method

The proposed unsupervised TTS system contains two modules: an alignment module to obtain pseudo-text for each utterance and a synthesis module that trains on pseudo-text. We evaluate the proposed unsupervised TTS system in English, as well as in six other languages (Hungarian, Spanish, Finnish, German, and Japanese).

3.1. Alignment Module

Motivated by wav2vec-U [8], our alignment module greedily proposes a pairing relationship (which we denote as an alignment) between real speech utterances and pseudo-transcripts. The alignment module follows a two-step approach: GAN training and self-training. In the GAN training step, a 1-layer CNN acts as the generator, which takes the segment representations extracted from a pre-trained wav2vec 2.0 model [22] and outputs a sequence of distributions over text units, where consecutive segments with the same argmax value are collapsed. The discriminator, a 3-layer CNN trained against the generator, tries to tell which source (real or generated) the input sequence is from. This is achieved by iteratively maximizing the likelihood of the generated phoneme sequence to train the generator and minimizing the binary cross-entropy loss to train the discriminator. In addition, since GAN training can be very unstable, we search over the weights for regularization losses such as gradient penalty loss, segment smoothness penalty, and phoneme diversity loss as described in [8]. We also validate the model with 50-100 transcribed utterances from the corpus to ensure convergence instead of using the unsupervised metric in [8]. After GAN training, greedy decoding is applied to the generator's output over the training set. We then train a hidden Markov model (HMM) with framewise speech representa-

tions extracted from a wav2vec 2.0 model as input and pseudo-text decoded by the generator as output. Finally, we decode the entire corpus again using the newly-trained HMM to obtain pseudo-transcripts for the supervised TTS system. Except for English, we opt not to further fine-tune a wav2vec 2.0 model with the pseudo-transcripts from the HMM.

3.2. Synthesis Module

The synthesis module uses the proposed alignment pairs from the alignment module and learns to synthesize speech from pseudo-text. Our synthesis module is motivated by Tacotron 2 [2] with guided attention loss [23]. During training, we perform an unsupervised model selection process by feeding the module with pseudo-transcripts when computing validation loss. During evaluation, ground truth transcripts are used as inputs to the synthesis module. Character error rates (CER) and word error rates (WER) are used to measure how much linguistic content is preserved by the synthesis module. We train a fully supervised TTS system using real text instead of pseudo-text and calculate the CER and WER on the same subset for comparison. To obtain the CER and the WER on each language, we either directly use a publicly available wav2vec 2.0 speech recognizer (for English) or fine-tune a pre-trained wav2vec 2.0 model on each language individually.

4. Experiments

4.1. Unsupervised TTS on English

We first evaluated the two-stage unsupervised TTS system on English. To train the alignment module, we used speech utterances from the 24-hour single-speaker LJSpeech corpus [24] and text samples from the LibriSpeech language modeling corpus [25]. We set aside about 300 utterances for validation and about 500 utterances for testing. We kept the ground truth transcripts for validation and test sets and used the rest for training without ground truth transcripts. The speech representations were extracted using a publicly available wav2vec 2.0 Large model trained on LibriLight [26], and the segment representations were built following the pre-processing procedures in [8].

The non-parallel text samples used for training, as well as the ground truth transcripts for the validation and test utterances, were converted to phones using a grapheme-to-phoneme (G2P) converter [27]. The best weights for the auxiliary penal-

Table 1: *Alignment module results on the LJSpeech dataset using English wav2vec 2.0 pre-trained features*

Language	Duration (hr)	Unsup ASR (PER)	
		No ST	ST
English	24	12.37	3.59

Table 2: *Unsupervised TTS results on the LJSpeech dataset using English wav2vec 2.0 pre-trained features*

Language	Unsup TTS		Supervised TTS	
	CER	WER	CER	WER
English	4.56	11.95	3.93	10.76

Figure 2: *Mel-spectrograms for ground truth (upper) and synthetic speech by the unsupervised TTS model (lower) for the English sentence "in being comparatively modern."*

ties of the GAN system, i.e., code penalty, gradient penalty, and smoothness weight, c.f. [8], were determined by grid search, and we chose the best model based on its PER over the validation set after 150k steps with a batch size of 160. GAN training is sometimes unstable in ways that we could only detect by using 50-100 supervised validation examples, which were the only places during training where we used paired data. The results of this stage is shown in Table 1. After determining the best GAN model, its output phone sequence was then refined using a self-training (ST) process [8] as follows. First, we used framewise wav2vec 2.0 features after PCA transformation as input and pseudo phone sequences transcribed by the generator as targets to train a triphone HMM. The triphone output from the HMM was decoded into words with an HCLG decoding graph, and we further fine-tuned a wav2vec 2.0 Large model using the pseudo character targets obtained from the above step, under the Connectionist Temporal Classification (CTC) loss [28]. Both steps were validated with the corresponding pseudo-text for the validation set. As shown in Table 1, ST reduces the phone error rate on the test set by 70% relative and provides very accurate transcripts for the second-stage TTS system. We used the publicly available wav2vec-U model in the Fairseq toolkit [29] to train the GAN and used the Kaldi toolkit [30] to train the triphone HMM and to build the decoding graph.

To train the synthesis module, we used the Tacotron 2 [2] implementation in ESPnet [31]. The ESPnet implementation follows the original Tacotron 2 model, except that another guided attention loss [23] was calculated on top of the encoder-decoder attention matrix so that it is not too far from being diagonal. During training, the synthesis module takes pseudo phone transcripts as inputs, and outputs 80-dimensional mel-spectrograms. These pseudo phone transcripts are converted by G2P from the word-level hypotheses generated by the fine-tuned wav2vec 2.0 model (in the final step of alignment module training). The synthesis module was trained for 80 epochs, with the same validation and test splits used for training the alignment module. During validation of the synthesis module, we calculated the reconstruction loss based on pseudo-text instead of real text. During testing, we fed the trained synthesis module with real, phonemicized text transcripts for the test set to obtain mel-spectrograms and synthesized raw audios with HiFi-GAN [32]. We calculated the CERs and raw WERs without additional language models using a publicly available wav2vec 2.0 Large model fine-tuned on LibriSpeech. Table 2 shows the two error rates on the synthesized test utterances
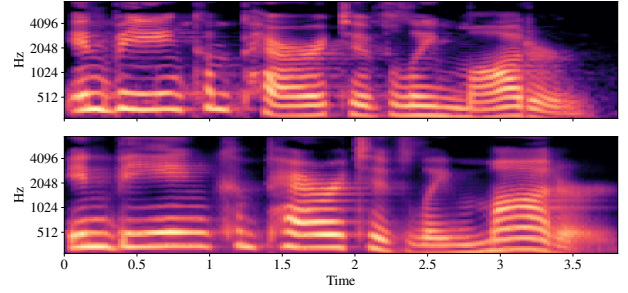
using our proposed unsupervised system (Unsup TTS). Compared with a fully-supervised Tacotron 2 model trained and validated with real, phonemicized text transcripts, our unsupervised system only lags behind 0.63% absolute in terms of CER and 1.19% absolute in terms of WER. Figure 2 plots the mel-spectrogram of a synthetic speech example by our unsupervised model, which shows that except for the temporal patterns, the mel-spectrogram by the unsupervised TTS looks very similar to the ground truth with very little loss of linguistic content.

4.2. Unsupervised TTS on CSS10 Languages

Table 3: *Unsupervised ASR results on the CSS10 dataset using English wav2vec 2.0 pretrained features*

Language	Duration (hr)	Unsup ASR (CER)	
		No ST	ST
Japanese	15	26.12	17.81
Hungarian	10	25.08	15.26
Spanish	24	20.80	14.57
Finnish	10	29.78	21.00
German	17	26.31	19.47
Dutch	14	45.65	39.24

Table 4: *Unsupervised TTS results on the CSS10 dataset using English wav2vec 2.0 pretrained features*

Language	Unsup TTS		Supervised TTS	
	CER	WER	CER	WER
Japanese	17.98	47.81	17.87	36.23
Hungarian	27.78	76.82	18.05	63.14
Spanish	23.03	55.52	18.19	36.74
Finnish	36.05	84.46	22.84	58.67
German	17.25	56.78	11.28	40.94
Dutch	53.01	89.41	34.53	76.71

We evaluated our unsupervised TTS system on six additional languages: Japanese, Hungarian, Spanish, Finnish, German and Dutch from the CSS10 dataset [33]. The total duration of each language is listed in Table 3. The experiments followed similar steps as the English experiment in Sec 4.1. The align-

Table 5: *The effect of different pretrained vocoders (Griffin-Lim, HiFi-GAN) on unsupervised TTS results for LJSpeech and various languages from CSS10*

Language	Griffin-Lim		HiFi-GAN	
	CER	WER	CER	WER
English	5.02	12.83	**4.56**	**11.95**
Japanese	**17.98**	**47.81**	20.58	54.09
Hungarian	27.78	76.82	**26.92**	**76.60**
Spanish	**23.03**	**55.52**	29.41	68.82
Finnish	**36.05**	**84.46**	37.66	87.48
German	**17.25**	**56.78**	18.45	59.90

Table 6: *The effect of different text units on unsupervised TTS using Griffin-Lim vocoder*

Language	Phoneme		Grapheme	
	CER	WER	CER	WER
Hungarian	**22.73**	**68.80**	27.78	76.82
Finnish	**27.58**	**67.87**	36.05	84.46
Dutch	**22.04**	**56.85**	53.01	89.41

ment module extracts speech representations from the same English wav2vec 2.0 Large model, followed by GAN training and self-training. The results for the alignment module are shown in Table 3. There were still a few differences in details in this multilingual experiment. Due to resource limits, these multilingual experiments used a potentially easier setting where both the audio and text were drawn from the same CSS10 dataset with their paired relationship broken up, instead of from different datasets as for English. We did not convert graphemes into phonemes and directly used the characters in each language as text unit. We split the audio and text data into training and validation sets with a ratio of 99 to 1, leaving about 50 to 100 validation utterances depending on the dataset size. The self-training step of the first stage only contained a character-based HMM (instead of a triphone HMM with HCLG decoding) for generating pseudo-text, and we did not have a second step of fine-tuning a wav2vec 2.0 model as for English. During the evaluation of the synthesis module, we used a Griffin-Lim vocoder to synthesize the audios from the generated mel-spectrogram, and the results reported in Table 4 were calculated using audios from the Griffin-Lim vocoder instead of the HiFi-GAN vocoder. We switched to the Griffin-Lim vocoder because we empirically found that it yielded lower error rates on these languages. To calculate CER and raw WER, we fine-tuned a publicly available wav2vec 2.0 Base model for each language individually, using paired speech and character-level transcripts from each CSS10 corpus.

The multilingual results in Table 4 confirm the conclusions we reach in the English experiments. Although the self-training step is simplified to only a character-based HMM, the self-training step still greatly reduces the error rates by 25% to 40% relative to all the languages. Compared to the fully-supervised Tacotron 2 models trained using real text transcripts, the CERs of our unsupervised systems differ by only about 9% absolute on average while requiring only a few paired utterances during validation. Further, we observe that the gap in WER between supervised and unsupervised TTS systems generally is about

10-20% absolute for all languages except Finnish, a much larger gap than CER. We hypothesize that it may be due to the lack of a robust language model in the TTS systems, making it harder for the model to preserve word-level information when training with noisy (pseudo-)transcripts. Last but not least, we observe that the performance of the alignment module does not always limit the performance of unsupervised TTS. In the case of German, the TTS trained with pseudo-transcripts achieves a *lower* CER compared to the alignment module alone, which suggests that the TTS has some internal mechanism to correct the noise in the pseudo-transcripts.

4.3. Comparison Between Griffin-Lim and HiFi-GAN

A comparison between the error rates of using Griffin-Lim and HiFi-GAN vocoders is presented in Table 5. We observe that the Griffin-Lim vocoder yields lower CERs and WERs than the HiFi-GAN vocoder in all languages except English and Hungarian, even though informal listening suggests that HiFi-GAN generates more natural speech with fewer artifacts. We hypothesize that HiFi-GAN works better for English because it is pretrained on the English LJSpeech dataset and may not generalize very well when applied to datasets of different languages.

4.4. Comparison Between Phoneme and Grapheme

We trained additional phoneme-based unsupervised TTS models in Hungarian, Finnish, and Dutch to study how the text units affect system performance. The training procedure was the same as that described in Sec 4.2, except that for training the alignment module, we converted the language-specific graphemes to the phonetic annotations, i.e., the International Phonetic Alphabet, using LanguageNet G2Ps [34]. We then use the phonetic outputs from the HMM within the alignment module to train the synthesis module. The CERs and WERs are reported in Table 6. The table shows that the phone-based systems yield significantly lower error rates than the grapheme systems. As graphemes are the smallest functional unit of a writing system, it involves extra complexity on top of the phone systems. Thus, modeling the grapheme systems is harder than modeling the phone systems, as indicated by its higher error rates. The gap between grapheme and phoneme systems is considerably smaller for Hungarian and Finnish than for Dutch. One probable explanation is that spelling and phonetic transcription is far more regular for the former two languages than for Dutch.

5. Conclusions

In this work, we combined an alignment module and a synthesis module to build a unsupervised TTS system that trains without paired data. The final unsupervised TTS system demonstrates competitive intelligibility in English and a slight degradation in intelligibility in six other languages on the level of supervised TTS models. We further show that phonemes work better than graphemes as text units for our systems. In the future, we would like to explore unsupervised TTS with truly non-parallel datasets for languages other than English and ways to improve the stability for the alignment module.

6. Acknowledgements

7. References

[1] Y. Wang, D. S. RJ Skerry-Ryan, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio *et al.*, "Tacotron: Towards end-to-end speech synthesis," in *arXiv*, 2017. [Online]. Available: preprintarXiv:1703.10135

[2] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu1, "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *ICASSP*, 2018.

[3] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: 2000-speaker neural text-to-speech," in *ICLR*, 2018.

[4] 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*, 2019.

[5] N.Li, S.Liu, Y.Liu, S.Zhao, and M.Liu, "Neural speech synthesis with transformer network," in *AAAI*, vol. 33, 2019, p. 6706–6713.

[6] J. Xu, X. Tan, Y. Ren, T. Qin, J. Li, S. Zhao, and T. Liu, "LR-Speech: Extremely low-resource speech synthesis and recognition," in *KDD*, 2020, pp. 2802–2812.

[7] K. Park and T. Mulc, "CSS10: A collection of single speaker speech datasets for 10 languages," *Interspeech*, 2019.

[8] A. Baevski, W.-N. Hsu, A. Conneau, and M. Auli, "Unsupervised speech recognition," in *Neural Information Processing Systems*, 2021.

[9] P. K. Muthukumar and A. W. Black, "Automatic discovery of a phonetic inventory for unwritten languages for statistical speech synthesis," in *ICASSP*, 2014, pp. 2594–2598.

[10] A. H. Liu, T. Tu, H. Lee, and L. Lee, "Towards unsupervised speech recognition and synthesis with quantized speech representation learning," in *ICASSP*, 2020, pp. 7259–7263.

[11] H. Zhang and Y. Lin, "Unsupervised learning for sequence-to-sequence text-to-speech for low-resource languages," in *Interspeech*, 2020, pp. 3161–3165.

[12] K. Lakhotia, E. Kharitonov, W.-N. Hsu, Y. Adi, A. Polyak, B. Bolte, T.-A. Nguyen, J. Copet, A. Baevski, A. Mohamed, and E. Dupoux, "On generative spoken language modeling from raw audio," in *arXiv*, 2021. [Online]. Available: https://arxiv.org/pdf/2102.01192.pdf

[13] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhotia, W.-N. Hsu, A. Mohamed, and E. Dupoux, "Speech resynthesis from discrete disentangled self-supervised representations," in *arXiv*, 2021. [Online]. Available: https://arxiv.org/pdf/2104.00355.pdf

[14] E. Kharitonov, A. Lee, A. Polyak, Y. Adi, J. Copet, K. Lakhotia, T.-A. Nguyen, M. Rivière, A. Mohamed, E. Dupoux, and W.-N. Hsu, "Text-free prosody-aware generative spoken language modeling," in *arXiv*, 2021. [Online]. Available: https://arxiv.org/pdf/2109.03264.pdf

[15] M. Hasegawa-Johnson, A. Black, L. Ondel, O. Scharenborg, and F. Ciannella, "Image2speech: Automatically generating audio descriptions of images," in *ICNLSSP*, 2017, p. 1–5.

[16] X. Wang, S. Feng, J. Zhu, M. Hasegawa-Johnson, and O. Scharenborg, "Show and speak: directly synthesize spoken description of images," in *icassp*, 2021.

[17] W.-N. Hsu, D. Harwath, T. Miller, C. Song, and J. Glass, "Text-free image-to-speech synthesis using learned segmental units," in *ACL-IJCNLP*, 2021, pp. 5284–5300.

[18] J. Effendi, S. Sakti, and S. Nakamura, "End-to-end image-to-speech generation for untranscribed unknown languages," *IEEE Access*, vol. 9, pp. 55 144–55 154, 2021.

[19] Y. Ren, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Almost unsupervised text to speech and automatic speech recognition," in *ICML*, 2019, pp. 5410–5419.

[20] C.-K. Yeh, J. Chen, C. Yu, and D. Yu, "Unsupervised speech recognition via segmental empirical output distribution matching," in *ICLR*, 2019.

[21] K.-Y. Chen, C.-P. Tsai, D.-R. Liu, H.-Y. Lee, and L. shan Lee, "Completely unsupervised speech recognition by a generative adversarial network harmonized with iteratively refined hidden Markov models," in *Interspeech*, 2019.

[22] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Neural Information Processing Systems*, 2020.

[23] H. Tachibana, K. Uenoyama, and S. Aihara, "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention," in *ICASSP*, 2018, pp. 4784–4788.

[24] K. Ito and L. Johnson, "The lj speech dataset," https://keithito.com/LJ-Speech-Dataset/, 2017.

[25] 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.

[26] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P. E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux, "Libri-light: A benchmark for asr with limited or no supervision," in *ICASSP*, 2020, pp. 7669–7673.

[27] K. Park and J. Kim, "g2pe," https://github.com/Kyubyong/g2p, 2019.

[28] A. Graves, S. Fernández, F. J. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *ICML*, 2006, pp. 369–376.

[29] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, "fairseq: A fast, extensible toolkit for sequence modeling," in *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.

[30] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *ASRU*, 2011.

[31] T. Hayashi, R. Yamamoto, K. Inoue, T. Yoshimura, S. Watanabe, T. Toda, K. Takeda, Y. Zhang, and X. Tan, "ESPnet-TTS: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit," in *ICASSP*, 2020, pp. 7654–7658.

[32] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," in *Neural Information Processing Systems*, 2020.

[33] K. Park and T. Mulc, "Css10: A collection of single speaker speech datasets for 10 languages," in *Interspeech*, 2019.

[34] M. Hasegawa-Johnson, L. Rolston, C. Goudeseune, G.-A. Levow, and K. Kirchhoff, "Grapheme-to-phoneme transduction for cross-language asr," in *SLSP*, 2020, pp. 3–19.