INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Tone Learning in Low-Resource Bilingual TTS

Ruolan Liu, Xue Wen, Chunhui Lu, Xiao Chen

Samsung Research China-Beijing(SRC-B)
{ruolan.liu, xue.wen, chunhui.lu, xiao.chen}@samsung.com

Abstract

We present a system for low-resource multi-speaker cross-lingual text-to-speech synthesis. In particular, we train with monolingual English and Mandarin speakers and synthesize every speaker in both languages. The Mandarin training data is limited to 15 minutes of speech by a female Mandarin speaker. We identify accent carry-over and mispronunciation in low-resource language as two major challenges in this scenario, and address these issues by tone preservation mechanisms and data augmentation, respectively. We apply these techniques to a recent strong multi-lingual baseline and achieve higher ratings in intelligibility and target accent, but slightly lower ratings in cross-lingual speaker similarity.

Index Terms: low resource, multilingual, speech synthesis, tone learning, tone preservation

1. Introduction

Multilingual text-to-speech (ML-TTS) aims to generate natural, intelligible speech in different languages while maintaining perceived speaker identity. It finds applications in all multilingual voice interfaces such as chatbot, satnav and speech-to-speech translation. ML-TTS is typically discussed under various data constraints to rule out trivial reduction to multiple independent monolingual synthesizers. For example, we will not have large data of a bilingual speaker in both languages.

We consider a bilingual scenario where only monolingual speakers are available and one language is low-resource. In particular, we select English and Mandarin Chinese, from different language families. English is treated as high-resource with 3 female and 3 male speakers and $\sim$12k utterances available for training. Mandarin is treated as low-resource with 333 utterances (15min) by one female speaker. Our goal is to synthesize all speakers in both languages, with particular interest in cross-lingual synthesis, i.e. Mandarin by native English speakers and English by native Mandarin speakers. We do not discuss code-mixed synthesis.

In this paper we address two challenges posed by our scenario: accent carry-over and mispronunciation in low-resource language. Accent carry-over refers to utterances synthesized in one language carrying the accent of another, typically that of the target speaker's native language (which we call the "source language" in this paper). This is often seen in human 2nd-language learners and is closely related to the reproduction of one's native prosodic patterns in 2nd language. Mispronunciation refers to the synthesizer being unable to produce the correct phonetic sequence. This is particularly relevant to low-resource synthesis, where the synthesizer does not see enough examples to learn proper phonetization. Our preliminary studies observe both accent carry-over and mispronunciation worsen at low resource availability. We address accent carry-over by two tone preservation mechanisms that help retaining source tone information in the dataflow: one using an auxiliary tone predicting regularizer, the other directly injecting tone representations half way. We address mispronunciation by data augmentation, using noise and speed perturbation to provide more, if not independent, data for learning the low-resource language. We apply these techniques to a strong end-to-end baseline and achieve consistent improvements in accent and intelligibility evaluations, at the cost of slight loss of cross-lingual speaker similarity.

Section 2 briefly summarizes recent related work in multilingual TTS. Section 3 presents our methods in detail. Section 4 describes our evaluation setup and reports results.

2. Related Work

2.1. Multilingual speakers

A straightforward way to do ML-TTS is using multilingual speech from a multilingual speaker. [1] presented a Mandarin-English TTS system that shared hidden Markov model (HMM) states between languages, using recordings from a bilingual speaker. [2] proposed a speaker-language factorization method in deep neural network (DNN) based TTS using three bilingual speakers. [3] learned to transform speaker embedding between languages from a bilingual speaker, then applied to other monolingual speakers.

2.2. Multilingual synthesis from monolingual speakers

Professional-level multilingual speakers are rare and collecting multilingual speech in quantity is expensive. A train of researches turned to easily accessible large monolingual corpora. [4–7] investigated combining monolingual speech from different languages and speakers for multilingual parametric TTS. As each speaker speaks only one language, speaker and language characteristics are highly correlated. This may lead to heavy accent carry-over in synthesized speech, or inconsistent voice between languages.

2.3. Multilingual end-to-end

Recent progress in end-to-end monolingual TTS [8–12] prompted studies to extend these systems to the multilingual task [13–17]. [14] used Unicode bytes to unify the text input format across languages. Their system was trained on 127 hours of speech and was capable of reading code-switched text, but suffered from cross-language speaker inconsistency. [15] trained an English-German-Spanish TTS with over 400 speakers. It used a speaker preserving loss to improve the speaker consistency across languages. [17] trained multilingual TTS with 550 hours of speech data from 92 speakers, using an adversarial speaker loss to disentangle speaker from language.

2.4. Low-resource TTS

A majority of languages in the world remain low-resource for various difficulties in data collection and labelling. Construction TTS for such languages often require different techniques.

http://dx.doi.org/10.21437/Interspeech.2020-2180

[18, 19] achieved low-resource statistical TTS by adapting a parametric model trained on multiple languages to a new low-resource language. [20] applied a similar method to end-to-end TTS. The authors used data in a high-resource language to pre-train their TTS engine, then adapted it to low-resource languages.

In this paper we look at both the multilingual and low-resource aspects of the problem.

3. Methods

3.1. Baseline

Our baseline synthesizer is adapted from [17], a multilingual neural TTS based on Tacotron 2 [9], as shown in Figure 1. It uses an attentional encoder-decoder model as the backbone, an adversarially-trained speaker classifier to disentangle speaker from language, a variational-autoencoder-like "residual encoder" to improve stability, and a post-processing net to convert mel spectrogram to linear spectrogram. Speaker and language embeddings are injected at decoder input. Griffin-Lim [21] is used to construct audio output.

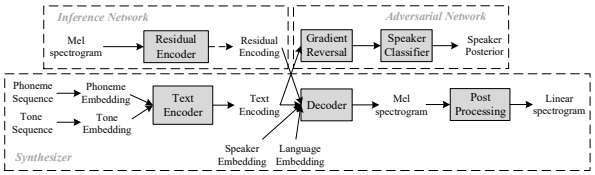


Figure 1: *Baseline architecture*

We adapt this system to our specific setup as follows.

3.1.1. Input representation

Our synthesizer takes phoneme-tone sequence as inputs. No phoneme is shared across languages. All double vowels and nasal vowels in Mandarin are divided into two or three single vowels. For tone embedding we use four pitched tones and one "toneless" tone for Mandarin, "stressed" and "unstressed" tones for English.

3.1.2. Speaker representations

We use x-vectors [22] for speaker embedding, which have been successfully applied to multi-speaker TTS [23]. We train our x-vectors following [22] with large amount of training data in Mandarin and English. We argue this does not severely breach low-resource assumption on Mandarin, as learning x-vectors needs only speaker labels, which are cheaper than text labels. When training the TTS model, we extract x-vector from each training utterance for speaker embedding. At synthesis time the target speaker's mean x-vector during training is used.

Our preliminary experiment with speaker embeddings learned within ML-TTS showed a tendency to render Mandarin speech in the voice of the Mandarin speaker, rather than the expected target speaker. We speculate this related to the synthesizer seeing only one Mandarin speaker during training. X-vectors learned on large number of speakers should relieve this problem.

3.1.3. Residual encoder

The structure of the residual encoder is the same as that in [17]. The variational autoencoder-like residual encoder computes latent factors of the audio during training phase. The prior mean (all zeros) is fed at synthesis time. We observe that, although training data of a speaker is noisy, clean speech can also be generated just by feeding all zeros to decoder, which makes data augmentation by adding noise possible.

3.1.4. Training objective

The objective function can be formulated as combining an evidence lower bound (ELBO) with a domain adversarial training objective:

$$\begin{aligned}&\mathcal{L}_1(\theta, \phi_r; speech, text, \mathbf{y}_s)\\&= ELBO(\theta, \phi_r; speech, text) - \lambda_s \mathcal{L}_2(\psi_s; text, \mathbf{y}_s)\end{aligned} \tag{1}$$

θ, ϕ_r and ψ_s are parameters of the synthesizer, residual encoder and adversarial speaker classifier, respectively, and $\mathbf{y}_s$ is the speaker label. Our ELBO is actually a β-VAE objective [24] under standard Gaussian latent prior:

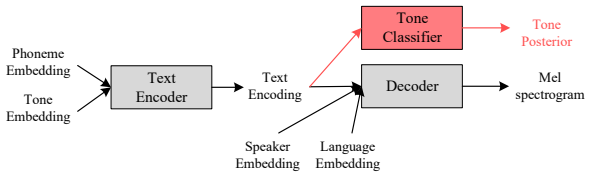
$$\begin{aligned}&\mathbb{E}_{q(\mathbf{z}_r|speech)}[\log p(speech|\mathbf{z}_r, text)]\\&- \lambda_{KL} D_{KL}(q(\mathbf{z}_r|speech)) \| \mathcal{N}(0, I))\end{aligned} \tag{2}$$

We will use $0 < \lambda_{KL} < 1$, which favours accuracy over latent space exploration.

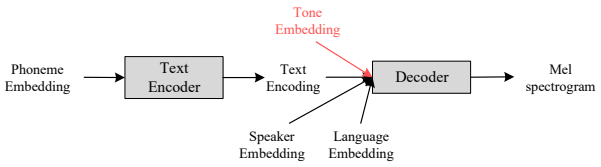
3.2. Tone preservation mechanisms

How does accent carry-over happen in our synthesizer? Obviously the text input contains full information of target accent and no information of source accent. Given that such carry-over is rarely observed when source and target languages are the same, we suspect that source accent has entered via speaker embedding during cross-lingual synthesis, and competes with target accent at the input layer of the decoder. Accent carry-over happens if source accent wins.

During training of the system the target speaker is always native in the target speech. Both encoder output and speaker embeddings provide information to construct correct prosodic accent of target speech. This gives the speaker embeddings a chance to "explain-away" some prosodic information from encoder output, so that during cross-lingual synthesis the decoder receives weakened target accent information from encoder. In this subsection we introduce two simple, independent mechanisms for tone preservation. They work on the principle of strengthening target tone information at decoder input.



(a) Implicit tone preservation (ITP)



(b) Explicit tone representation (ETR)

Figure 2: *Tone preservation mechanisms*

3.2.1. Implicit tone preservation (ITP)

We add an auxiliary classifier that predicts tone labels from encoder output, as shown in Figure 2(a), and train it along with the TTS engine via a regularizer:

$$\mathcal{L}(\theta, \phi_r, \psi_t; speech, text, \mathbf{y}_s, \mathbf{y}_t) = \mathcal{L}_1(\theta, \phi_r; speech, text, \mathbf{y}_s) + \lambda_t \mathcal{L}_3(\psi_t; text, \mathbf{y}_t) \quad (3)$$

where ψ_t parameterizes the tone classifier and $\mathbf{y}_t$ is the tone label. Doing so promotes preservation of target tone information throughout the encoder. Speaker embedding does not contribute to this classifier therefore has no path to explain this information away. We call this scheme implicit tone preservation.

We use a two-layer feedforward network for tone classification and standard cross-entropy loss for $\mathcal{L}_3$.

3.2.2. Explicit tone representation (ETR)

We move target tone input from encoder input to decoder input, as shown in Figure 2(b). Doing so directly provides the decoder with strong target accent information that cannot be explained away. We call this scheme explicit tone representation.

Similar treatment is also found in expressive TTS, which often feeds a style embedding to the decoder [25,26] to generate different expressions of the same text. The analog is relevant as both tone and style encode prosodic modulation characteristics.

We use either ITP or ETR in our synthesizer, but not both together.

3.3. Data augmentation

Data augmentation is a common strategy training large model on small data. We apply 10-fold data augmentation to the training set of low-resource language by noise and speed perturbations. For each training example we create 4 additional versions speed-perturbed to 80%, 90%, 110% and 120% of original rate and tag them as 4 new speakers. Vehicle noise is then added to all examples at SNR 0dB to double the data size.

4. Experiments

4.1. Setup

4.1.1. Data

Our experiments are conducted on an internal American English and Mandarin Chinese datasets, with no bilingual speaker. The English training set has 6 English speakers, 3 male and 3 female, about 12,000 utterances in total. The Mandarin training set has one female Mandarin speaker, 333 utterances in total (15min). All utterances come with transcriptions as phoneme-tone sequences.

Recordings are sampled at 24k Hz. 80-dimension mel-scale and 1025-dimension linear-scale spectrograms are extracted every 10ms.

4.1.2. Details

We use >200 hours of speech data from 300 Mandarin and American English speakers to train our x-vector extractor using Kaldi toolkit [27]. The size of x-vectors is set at 64.

Details of the baseline model is the same as [17], except that we set λ_{KL} to 0.2, which we find beneficial for generating clean speech. Training the TTS engine took about 150k steps at batch size 32 on one P40 GPU. Different methods are compared in our experiments. The configuration of each method is described as follows.

- **Implicit tone preservation (ITP):** The auxiliary tone classifier has two feedforward layers and hidden layer size 256. We set λ_t to 0.2.
- **Explicit tone representation (ETR):** We concatenate tone, speaker and language embeddings and feed them to the decoder along with encoder output.
- **Data augmentation (DA):** Noise adding and speed perturbation are applied for Mandarin. We use SoX tool [28] for speed perturbation.

4.2. Subjective evaluation

We evaluate our proposed techniques on a strong multilingual baseline adapted from [17]. Comparisons are made between baseline (Base), baseline+ITP+DA (ITP-DA) and baseline+ETR+DA (ETR-DA).

4.2.1. MOS

We first investigate mean option score (MOS) in speaker similarity, intelligibility and target accent [4]. All speakers in both languages are evaluated. 20 native Mandarin speakers are asked to listen to the generated utterances and rate them on a scale between 1 and 5. The results are given in Table 1.

All three synthesizers achieve similar speaker similarity when synthesizing target speaker's native language. Slightly lower score is observed in Mandarin, we may be attributed to insufficient training data. For cross-lingual synthesis we find our methods hurt similarity MOS by ~ 0.25. We suspect this is related to entanglement of speaker identity with preserved target tone information, which was suppressed by the adversarial regularizer in the baseline.

The comparison of the performance on intelligibility shows that all three methods achieve consistent and good performance when the source and target language are the same. Mandarin gets slightly higher scores probably due to the raters being native Mandarin speakers. For cross-language synthesis, the baseline has a large gap behind the proposed methods, which illustrates the effectiveness of the latter.

For target accent, we observe that the results are very similar to those on intelligibility. We suppose that target accent and intelligibility have an interdependence with each other and the proposed methods improve the performance of the both.

4.2.2. AB preference test

We run A-B preference tests to evaluate effects of tone preservation and data augmentation separately. For each test we present the rater with two speech stimuli, one synthesized with ETR (or DA) and one without, and ask the rater to choose a preferred one.

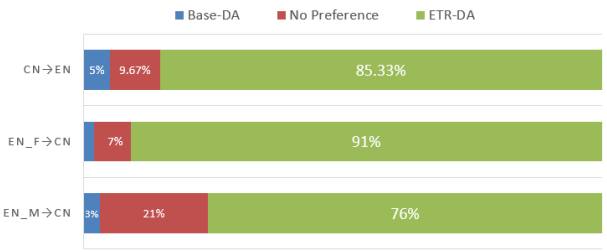


Figure 3: *AB preference results of Base-DA and ETR-DA on tone accuracy for cross-language synthesis*

Table 1: *Comparison of MOS on speaker similarity, intelligibility and target accent*

Source Language	Method	Speaker similarity		Intelligibility		Target accent	
		CN	EN	CN	EN	CN	EN
CN	Base	4.08 ± 0.17	3.60 ± 0.25	4.97 ± 0.00	3.01 ± 0.31	4.94 ± 0.01	2.98 ± 0.31
	ITP-DA	4.13 ± 0.35	3.33 ± 0.35	4.96 ± 0.00	4.47 ± 0.10	4.93 ± 0.01	4.24 ± 0.07
	ETR-DA	4.09 ± 0.28	3.35 ± 0.37	4.98 ± 0.00	4.40 ± 0.15	4.91 ± 0.01	4.24 ± 0.16
EN-Female	Base	3.80 ± 0.26	4.55 ± 0.20	3.75 ± 0.26	4.70 ± 0.06	2.83 ± 0.11	4.71 ± 0.03
	ITP-DA	3.54 ± 0.25	4.64 ± 0.16	4.43 ± 0.12	4.64 ± 0.10	3.54 ± 0.10	4.71 ± 0.04
	ETR-DA	3.55 ± 0.31	4.50 ± 0.23	4.55 ± 0.09	4.67 ± 0.08	3.56 ± 0.11	4.71 ± 0.04
EN-Male	Base	3.86 ± 0.26	4.56 ± 0.14	3.61 ± 0.20	4.67 ± 0.05	2.90 ± 0.12	4.67 ± 0.03
	ITP-DA	3.59 ± 0.33	4.51 ± 0.15	4.20 ± 0.16	4.65 ± 0.07	3.26 ± 0.14	4.67 ± 0.04
	ETR-DA	3.62 ± 0.22	4.63 ± 0.17	4.22 ± 0.14	4.63 ± 0.08	3.29 ± 0.14	4.67 ± 0.04

Figure 3 compares Base-DA and ETR-DA on tone accuracy for cross-language synthesis. We see that ETR obtains significantly higher preference score on tone accuracy. We believe that utilizing ETR, strong target accent information can be provided for decoder directly to construct correct prosodic accent.

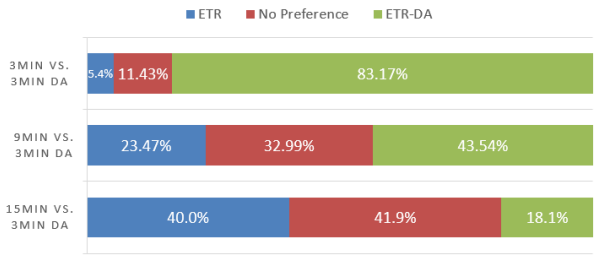


Figure 4: *AB preference results of ETR and ETR-DA on naturalness for Mandarin synthesis*

For data augmentation we evaluate synthesizers trained on smaller training sets of 3 minutes and 9 minutes, constructed as two random subsets of the 15 minutes. Figure 4 compares ETR-DA trained on 3 minutes' data and ETR trained on 3/9/15 minutes on naturalness of direct Mandarin synthesis. Performance of ETR-DA trained on 3 minutes' data is between those trained on 9 and 15 minutes without DA. This indicates significant potential of DA for improving quality on extremely-low-resource languages, as expected.

4.3. An objective score: tone preservation

Table 2: *Accuracy of tone classifiers trained on text encodings*

	Base-DA	ITP-DA
accuracy(%)	72.50	99.98

We quantify the degree of tone preservation described in 3.2.1 by classifying tones from encoder output. Two linear discriminative analysis classifiers are trained to predict tone class from text encodings of ITP-DA and Base-DA, respectively. The results are given in Table 2. ITP-DA achieves a higher tone prediction accuracy, implying that stronger tone information is maintained.

Figure 5: *t-SNE visualization of x-vectors extracted from speech synthesized by ETR-DA and Base-DA with different speakers and languages*

4.4. A visualization: speaker similarity

We use t-SNE [29] to visualize x-vectors computed from speech synthesized with ETR-DA and Base-DA, and give the result in Figure 5. When source and target languages are consistent, x-vectors from ETR-DA and Base-DA roughly overlap. These represent the "correct" positions of the speakers because they get high scores in similarity evaluation. When it comes to cross-lingual synthesis, the x-vectors of ETR-DA and Base-DA form two clusters, the former drifting further away from the "correct" positions, linking ETR to lower speaker similarity.

5. Conclusions

We build a Mandarin-English TTS engine where all training speakers are monolingual and Mandarin data is limited. Two methods of tone preservation are proposed to help generate utterance in proper prosodic accent of target language. Data augmentation is used to improve the quality of synthesis in low-resource language.

For future work, first, we consider building the training set of high-resource language with more speakers but less data for each speaker. Doing so reduces the difficulty of data collection while providing the synthesizer with various speaker identities and pronunciations. Then, we plan to continue investigating methods to improve speaker similarity further.

6. References

[1] Y. Qian, H. Liang, and F. K. Soong, "A cross-language state sharing and mapping approach to bilingual (mandarin-english) TTS," *IEEE Trans. Audio, Speech & Language Processing*, vol. 17, no. 6, pp. 1231–1239, 2009.

[2] Y. Fan, Y. Qian, F. K. Soong, and L. He, "Speaker and language factorization in dnn-based TTS synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5540–5544.

[3] S. Maiti, E. Marchi, and A. Conkie, "Generating multilingual voices using speaker space translation based on bilingual speaker data," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7624–7628.

[4] J. Latorre, K. Iwano, and S. Furui, "New approach to the polyglot speech generation by means of an hmm-based speaker adaptable synthesizer," *Speech Communication*, vol. 48, no. 10, pp. 1227–1242, 2006.

[5] H. Zen, N. Braunschweiler, S. Buchholz, M. J. F. Gales, K. Knill, S. Krstulovic, and J. Latorre, "Statistical parametric speech synthesis based on speaker and language factorization," *IEEE Trans. Audio, Speech & Language Processing*, vol. 20, no. 6, pp. 1713–1724, 2012.

[6] B. Li and H. Zen, "Multi-language multi-speaker acoustic modeling for LSTM-RNN based statistical parametric speech synthesis," in *Interspeech*, 2016, pp. 2468–2472.

[7] I. Himawan, S. Aryal, I. Ouyang, S. Kang, P. Lanchantin, and S. King, "Speaker adaptation of a multilingual acoustic model for cross-language synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7629–7633.

[8] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. V. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Interspeech*, 2017, pp. 4006–4010.

[9] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4779–4783.

[10] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. C. Courville, and Y. Bengio, "Char2wav: End-to-end speech synthesis," in *International Conference on Learning Representations (ICLR)*, 2017.

[11] W. Ping, K. Peng, A. Gibiansky, S. Ö. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: Scaling text-to-speech with convolutional sequence learning," in *International Conference on Learning Representations (ICLR)*, 2018.

[12] Y. Taigman, L. Wolf, A. Polyak, and E. Nachmani, "Voiceloop: Voice fitting and synthesis via a phonological loop," in *International Conference on Learning Representations (ICLR)*, 2018.

[13] Y. Cao, X. Wu, S. Liu, J. Yu, X. Li, Z. Wu, X. Liu, and H. Meng, "End-to-end code-switched TTS with mix of monolingual recordings," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6935–6939.

[14] B. Li, Y. Zhang, T. N. Sainath, Y. Wu, and W. Chan, "Bytes are all you need: End-to-end multilingual speech recognition and synthesis with bytes," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5621–5625.

[15] E. Nachmani and L. Wolf, "Unsupervised polyglot text-to-speech," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 7055–7059.

[16] L. Xue, W. Song, G. Xu, L. Xie, and Z. Wu, "Building a mixed-lingual neural tts system with only monolingual data," in *Interspeech*, 2019, pp. 2060–2064.

[17] Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Z. Chen, R. Skerry-Ryan, Y. Jia, A. Rosenberg, and B. Ramabhadran, "Learning to speak fluently in a foreign language: Multilingual speech synthesis and cross-language voice cloning," in *Interspeech*, 2019, pp. 2080–2084.

[18] Q. Yu, P. Liu, Z. Wu, S. Kang, H. Meng, and L. Cai, "Learning cross-lingual information with multilingual BLSTM for speech synthesis of low-resource languages," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5545–5549.

[19] A. Gutkin, "Uniform multilingual multi-speaker acoustic model for statistical parametric speech synthesis of low-resourced languages," in *Interspeech*, 2017, pp. 2183–2187.

[20] Y.-J. Chen, T. Tu, C. chieh Yeh, and H. yi Lee, "End-to-end text-to-speech for low-resource languages by cross-lingual transfer learning," in *Interspeech*, 2019, pp. 2075–2079.

[21] D. W. Griffin and J. S. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Trans. Audio, Speech & Language Processing*, vol. 32, no. 2, pp. 236–243, 1984.

[22] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.

[23] J. Williams, J. Rownicka, P. Oplustil, and S. King, "Comparison of speech representations for automatic quality estimation in multi-speaker text-to-speech synthesis," in *arXiv preprint arXiv:2002.12645*, 2020.

[24] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," in *International Conference on Learning Representations (ICLR)*, 2017.

[25] D. Stanton, Y. Wang, and R. J. Skerry-Ryan, "Predicting expressive speaking style from text in end-to-end speech synthesis," in *IEEE Spoken Language Technology (SLT)*, 2018, pp. 595–602.

[26] Y. Wang, D. Stanton, Y. Zhang, R. J. Skerry-Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *International Conference on Machine Learning (ICML)*, 2018, pp. 5167–5176.

[27] "Kaldi asr toolkit," https://kaldi-asr.org/.

[28] "Sox, audio manipulation tool," http://sox.sourceforge.net.

[29] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.