Interspeech 2022
18-22 September 2022, Incheon, Korea

Zero-Shot Foreign Accent Conversion without a Native Reference

Waris Quamer[1], *Anurag Das*[1],
John Levis[2], *Evgeny Chukharev-Hudilainen*[2], *Ricardo Gutierrez-Osuna*[1]

[1]Department of Computer Science and Engineering, Texas A&M University
[2]Department of English, Iowa State University

{quamer.waris, adas, rgutier}@tamu.edu , {jlevis, evgeny}@iastate.edu

Abstract

Previous approaches for foreign accent conversion (FAC) either need a reference utterance from a native speaker (L1) during synthesis, or are dedicated one-to-one systems that must be trained separately for each non-native (L2) speaker. To address both issues, we propose a new FAC system that can transform L2 speech directly from previously unseen speakers. The system consists of two independent modules: a translator and a synthesizer, which operate on bottleneck features derived from phonetic posteriorgrams. The translator is trained to map bottleneck features in L2 utterances into those from a parallel L1 utterance. The synthesizer is a many-to-many system that maps input bottleneck features into the corresponding Mel-spectrograms, conditioned on an embedding from the L2 speaker. During inference, both modules operate in sequence to take an unseen L2 utterance and generate a native-accented Mel-spectrogram. Perceptual experiments show that our system achieves a large reduction (67%) in non-native accentedness compared to a state-of-the-art reference-free system (28.9%) that builds a dedicated model for each L2 speaker. Moreover, 80% of the listeners rated the synthesized utterances to have the same voice identity as the L2 speaker.

Index Terms: Foreign accent conversion, zero-shot learning, many-to-many voice conversion

1. Introduction

Foreign accent conversion (FAC) [1] aims to transform non-native speech to have the accent (or pronunciation patterns) of a native speaker while retaining the speaker identity. The transformed synthetic voice is referred to as a “golden speaker”, and finds application in computer assisted pronunciation training [1, 2] for second-language (L2) learners. Since the golden-speaker speech has the same timbre (voice identity) as the L2 learner, it is an ideal target for L2 learners to practice pronunciation, instead of using speech from a native (L1) speaker. Apart from pronunciation training, other applications of FAC include personalized text-to-speech (TTS) synthesis [3, 4], movie dubbing [5], and improving automatic speech recognition (ASR) performance [6].

Previous approaches to FAC have two major limitations. First, most methods need a reference utterance from an L1 speaker during synthesis, which limits the system to sentences that have been pre-recorded by the L1 speaker. As a result, this can lead to poor transfer of speaker identity between source and target speakers due to information entanglement [7]. Recently, Zhao et al. [8] proposed a ”reference-free” method that transforms L2 utterances directly, but the approach requires building a dedicated one-to-one model for each pair of L1 and L2 speakers, which requires considerable amounts of data for each L2 speaker. To address both issues, we propose a new FAC system that is both reference-free and zero-shot [9]. In other words, the proposed system does not require a reference L1 utterance at inference time, and can be directly used to generate accent-conversions for unseen L2 speakers from a single utterance (zero-shot). Further, the system does not need re-training or fine-tuning for any of its models.

We split the task of reference-free FAC into two subtasks, pronunciation correction and voice conversion, which are handled respectively by a translator module and a synthesizer module. Both modules use a sequence-to-sequence (seq2seq) model as their backbone, and are trained independently. Utterances from L1 and L2 speakers are first transformed into bottleneck features (BNFs), a linguistic representation derived from phonetic posteriorgrams that captures the pronunciation pattern of the utterance [7, 10]. The translator converts BNFs from a L2 speaker’s utterance into the BNFs that would have been produced by an L1 speaker. This is achieved by training the translator using a parallel corpus of utterances from L1 and L2 speakers. The synthesizer module [11] is a many-to-many voice conversion system, trained in a non-parallel fashion on a corpus of multiple speakers, which produces a Mel-spectrogram from the BNFs and a speaker embedding. During inference, a L2 utterance is fed to the translator, and its output is passed to the synthesizer, conditioned on the speaker embedding of the same L2 speaker. The result is a Mel-spectrogram that captures the voice quality of the L2 speaker and the accent of an L1 speaker. Finally, to generate audio, we pass the obtained Mel-spectrogram through a WaveRNN neural vocoder [12].

2. Related work

Early approaches in FAC involved building an articulatory synthesizer for the L2 speaker to map the speaker’s articulatory trajectories (e.g., lips and tongue movements) into their acoustic features (e.g., Mel Cepstra) using several techniques including GMMs [13], unit-selection [14], and DNNs [15]. Then, the synthesizer was driven by articulatory trajectories from a L1 speaker to generate native-accented speech. To avoid the need to collect articulatory data, Aryal and Gutierrez-Osuna [16] proposed a FAC system that only used acoustic information. This model adapted the conventional voice-conversion approach, which pairs source and target frames via dynamic time wrapping (DTW), by replacing DTW with a technique that matched source-target frames based on their MFCC similarity after vocal tract length normalization. Later, Zhao *et al.* [7] refined the approach by using phonetic posteriorgrams (PPGs) to compute the similarity between pairs of source and target acoustic frames. These early approaches generated accent conversions on a frame-by-frame basis. More recent studies on voice [17, 18] and accent conversion [19, 20] have used seq2seq models, which can model segmental and prosody features simultaneously, resulting in better performance. A particular seq2seq architecture of interest is Tacotron [21], which was

10.21437/Interspeech.2022-10664

proposed for text-to-speech synthesis and led to significant improvement in the acoustic quality of synthesized speech. Thereafter, Tacotron2 [11] further improved acoustic quality through a novel architecture and a WaveNet vocoder. Jia *et al.* [22] extended Tacotron2 for voice cloning by conditioning a speaker embedding on the decoder.

Zero-shot learning has also been used for voice conversion [23, 24] and voice cloning [22, 25], leading to systems that can synthesize speech for arbitrary speakers unseen during training. Recently, Ding *et al.* [26] adopted this zero-shot learning approach for accent conversion. However, their system required a reference native speaker at synthesis time, which can be limiting. To our knowledge, only two prior studies have examined the problem of generating accent conversion directly from an L2 utterance, also known as "reference-free" accent conversion. Liu *et al.* [19] proposed a system that first extracts linguistic and speaker representations through independently trained ASR model and speaker encoder, respectively. Then, they use these representations to condition a multi-speaker TTS model, which finally generates native-accented speech. Their system suffers from two drawbacks. First, their ASR model needs to be fine-tuned on the target non-native speaker; second, as they suggest in their evaluation, their system does not faithfully capture the voice identity of the target speaker. Zhao *et al.* [8] also proposed a reference-free system. Their system is trained in two steps. First, they train a conventional accent-conversion model that uses reference L1 utterances as an input to generate golden-speaker utterances for a target L2 speaker. After this, the reference L1 utterances can be discarded. In a second step, they train a "pronunciation correction" model to map the L2 utterances into the golden-speaker utterances obtained in the first step. In this fashion, during inference the pronunciation-correction model directly converts L2 speech into the corresponding golden-speaker speech. In perceptual evaluations, their system outperformed that of Liu *et al.* [19]. However, the system had a major drawback in that it needed to build a dedicated model for each L1-L2 speaker pair, which in turn requires large amount ($\sim 100s$ utterances) of data from each L2 speaker, making it impractical for pronunciation-training at scale.

3. Methods

In this work, we proposed a reference-free zero-shot accent conversion system that can synthesize native-accented speech for an L2 speaker without the need of a parallel L1 utterance. The system is composed of five independently trained models: (1) an acoustic model that generates bottleneck features (BNFs), a speaker-independent linguistic representation from speech, (2) a speaker encoder designed to capture the voice identity of a speaker, (3) an accent encoder that captures the accent of a speaker, (4) a translator module, containing a seq2seq model that consumes the BNFs of an L2 utterance and the accent embeddings from an accent encoder, and generates the BNFs that would have been produced by an L1 speaker, and (5) a synthesizer seq2seq model, that takes BNFs and speaker embeddings as inputs and synthesizes a Mel-spectrogram for an arbitrary speaker.

The workflow of our proposed approach is illustrated in Figure 1. As described above, we train two independent seq2seq models. The translator seq2seq model draws inspiration from the task of machine translation, which in our scenario is translating a fine-grained linguistic representation (i.e., the BNFs) from a L2 utterance into that of an L1 utterance. We train the translator in a parallel fashion using a corpus of L2 speakers having various accents and a reference L1 speaker, whose accent and pronunciation patterns the translator seeks to model. To train the translator, we pair utterances for each L2 speaker with parallel utterances from an L1 speaker. Then, for each pair of utterances, we pass the L2 utterance through the acoustic model to extract BNFs, which we assume only contain linguistic information and are speaker independent; see Figure 1a. Next, we feed the L1 utterance through the accent encoder to obtain an accent embedding. Then, the L2 BNFs and L1 accent embedding are feed to the translator, which is trained to generate the L1 BNFs at the output. In contrast, the synthesizer is trained in a self-supervised manner to reconstruct the inputs at the outputs, so it does not require a parallel corpus of speech. As shown in Figure 1b, we generate BNFs for an utterance and feed then to the synthesizer, conditioned on the corresponding speaker embedding, which is obtained by passing this utterance through the speaker encoder. The synthesizer is then trained to reconstruct the corresponding Mel-spectrogram of the utterance. During inference, we extract BNFs from an L2 speaker and feed them to the translator; see Figure 1c. Next, we pass the output of the translator with the speaker embedding generated for the L2 speaker used in previous step to the synthesizer to obtain a native-accented Mel-spectrogram. Finally, this Mel-spectrogram is fed to a WaveRNN neural vocoder [12] to generate a time-domain waveform. Below, we describe in detail each component of our system.

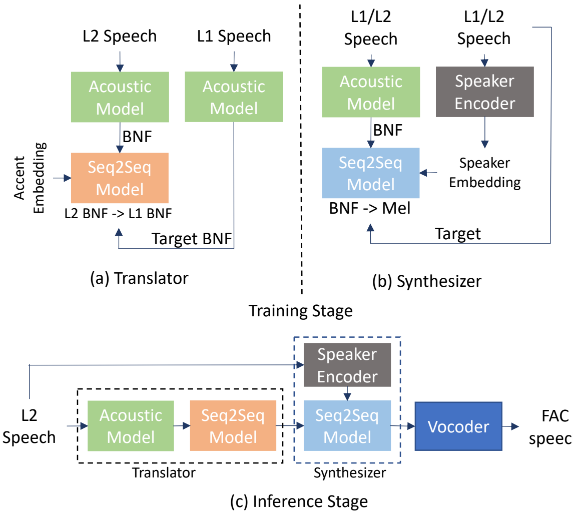

Figure 1: *Training workflow for (a) the translator and (b) the synthesizer. (c) Overall inference workflow of the system.*

3.1. Acoustic Model

Given an input utterance (L1 or L2), the acoustic model produces a phonetic-posteriorgram (PPG), which represents the posterior probability of each frame belonging to a predefined set of phonetic units (phonemes or triphones/senones). PPGs capture the linguistic content of an utterance and are assumed to be speaker independent. Following Zhao *et al.* [8], we use the output from the last hidden layer of the acoustic model (i.e., the layer prior to the final softmax layer) as bottleneck features (BNFs), instead of the PPGs. BNFs contain similar linguistic information as PPGs, but have much lower dimensionality (256 vs 6,024 for Senone-PPGs). We used the acoustic model described in [20].

3.2. Accent and Speaker Encoders

We use accent and speaker encoders to capture a speaker's accent and voice identity, respectively. The speaker encoder is trained as a speaker-verification model following the frame-

Table 1: *Parameters for our proposed accent conversion model.*

Component	Synthesizer Params	Translator Params
BNF	256 D	256 D
Speaker Emb.	256 D	-
Accent Emb.	-	256 D
3×Conv	512 5×1 kernels	512 5×1 kernels
2×pBiLSTM	256 cells/direction	256 cells/direction
Location Sensitive Attention	128-dim context; 32 31×1 conv kernels	512-dim context; 32 31×1 conv kernels
PreNet	2×FC; 256 units	2×FC; 512 units
2×LSTM	1024 cells	2048 cells
Linear (Mel/BNF)	1×FC; 80 units	1×FC; 256 units
Linear (stop-token)	1×FC; 1 unit	1×FC; 1 unit
PostNet	5×Conv; 512 5×1 kernel	5×Conv; 1024 5×1 kernel
Mel /BNF	80 D	256 D
Stop token	2 D	2 D

work in [27]. Given an utterance, the speaker encoder generates a fixed-dimension embedding vector, which represents the speaker identity. The speaker encoder model is composed of a 3-layer LSTM with 256 hidden nodes per layer. The hidden state of the last LSTM layer is fed to a projection layer with 256 units. We use the generalized end-to-end (GE2E) loss [27] for training, which maximizes the cosine similarity between utterances from the same speaker. The accent encoder follows the same architecture and paradigm as the speaker encoder but instead is trained to recognize various accents of English (see Section 4.)

3.3. Translator and synthesizer

The translator and synthesizer modules contains seq2seq models inspired by Tacotron2 [11]. During training, the inputs to the synthesizer are pairs of BNF sequences ($x \in \Re^{T \times D}$) and the corresponding speaker embeddings ($s \in \Re^{M}$). Here, T is the length of the BNF sequence, D is the BNF dimensionality (256 in this study), and M is the dimensionality of the speaker embedding (256 in this study). The encoder first takes in a sequence of BNFs x and generates a latent representation z as:

$$z = Encoder(x) \tag{1}$$

Then, we concatenate the latent representation z with the corresponding speaker embedding s to generate a concatenated representation:

$$z_{concat} = [z, s] \tag{2}$$

The concatenated latent representation z_{concat} is then fed to an attention mechanism to generate an attention context, which is then combined with z_{concat} and passed to the decoder to predict the Mel-spectrogram (o_{mel}) of the input speech in an autoregressive manner. This way the decoder is conditioned on identity of the target (L2) speaker.

$$o_{mel} = Decoder(z_{concat}) \tag{3}$$

We also pass the decoder output to a post-net which predicts the residual Mel-spectrogram, following [22]. We use the Euclidean distance between the target Mel-spectrogram and the model prediction before/after the post-net as the loss function. Simultaneously, we also minimize an additional cross-entropy loss to predict the stop-token, so that the generation process can be stopped during inference. The final loss function can be written as:

$$L = \alpha||o_{mel} - y_{mel}||_2^2 + \beta CE(o_{stop}, y_{stop}) \tag{4}$$

where, y_{mel} is the ground-truth Mel-spectrogram; y_{stop} is the ground truth stop token; $CE(.)$ is the cross-entropy loss and α, β are the weight terms.

The encoder in the original Tacotron2 architecture uses a bidirectional LSTM (Bi-LSTM) layer to process the input text embeddings. The input in our case are BNFs, which are significantly longer than text embeddings. Therefore, following Zhao et al. [8], we replace the Bi-LSTM layer in the encoder with two pyramidal Bi-LSTMs (p-Bi-LSTM) so that the high-level contextual and phonetic information can be captured from the input BNF sequence. With each p-Bi-LSTM, there is a two-factor reduction in the time resolution and hence our encoder generates four times shorter sequence as compared to the input. The Translator has the same architecture as the synthesizer, but the decoder is conditioned on the accent embedding instead of the speaker embedding and produces L1 equivalent BNF representations. The hyperparameters of each component are summarized in Table 1.

4. Experimental setup

We trained the acoustic model on the Librispeech [28] corpus, which contains utterances from 2,484 native English speakers. The acoustic model was implemented using Kaldi [29] and the trained model achieved a word error rate (WER) of 3.76% on Librispeech's test-clean subset. We used VoxCeleb1 [30], VoxCeleb2 [31], and Librispeech to train the speaker encoder, for a total of around 3,000 hours of speech from 9,847 speakers. Following [26], we trained the accent encoder using data from the Speech Accent Archive [32]. We extracted a 257-dimensional Mel-spectrogram with 25ms window and 10 ms shift. Both speaker and accent encoders were implemented in PyTorch and trained using the Adam optimizer with a learning rate of 10^{-2} and batch size of 128. For the FAC task, we trained both the seq2seq models and conducted experiments using the ARCTIC [33] and the L2-ARCTIC [34] corpora. For each speaker, we divided their utterances into three subsets: a training set of 1,032 utterances, a validation set of 50 utterances and a test set of 50 utterances. To replicate the experiments by Zhao *et al.* [8], four speakers (NJS, YKWK, TXHC, and ZHAA) from L2-ARCTIC were left out of the training set to run experiments as unseen speakers to the system. We use BDL as the reference native speaker to train the translator model and pair it with all 24 speakers from ARCTIC and L2-ARCTIC datasets (excluding the four test speakers and including BDL). The synthesizer model was also trained using these 24 speakers. We implement both seq2seq models using TensorFlow. All models were trained using two NVIDIA Tesla V100 GPUs. For both seq2seq models, we set the batch size as 32 and used Adam as the optimizer with a learning rate of 10^{-3} which was annealed down to 10^{-5} by exponential scheduling. The synthesizer and the translator converged after 200,000 and 300,000 steps, respectively, and in total took about 140 hours to train.

5. Results

To evaluate our proposed system, we conducted three perceptual experiments to rate three attributes of the synthesized speech: accentedness, acoustic quality, and voice similarity. We compare our proposed architecture to the reference-free system of Zhao *et al.* [8], which served as the baseline. For each perceptual experiment, we instructed participants to focus on the target speech attribute. Test utterances were randomly selected from the test set, and presentation order was randomized and counter balanced. We recruited 20 separate participants for each listening test. All participants were residents of the United

Table 2: *Mean Opinion Score rating scale.*

Rating	Speech Quality	Level of distortion
5	Excellent	Imperceptible
4	Good	Just perceptible but not annoying
3	Fair	Perceptible but slightly annoying
2	Poor	Annoying but not objectionable
1	Bad	Very annoying and objectionable

Table 3: *Accentedness rating (the lower, the better) and MOS (the higher, the better) for the two accent conversion systems and the original speech.*

	Baseline	Proposed	Original L2	Original L1
Accentedness	4.82	2.23	6.78	1.10
MOS	3.03	3.03	4.44	4.63

States at the time of the recruitment, and were required to pass a qualification test where they were asked to identify various regional dialects of the United States. The study was approved by the Institutional Review Board of Texas A&M, and was conducted on Amazon Mechanical Turk.

5.1. Accentedness test

Participants were asked to rate the degree of foreign accentedness of each audio sample on a nine-point Likert scale (1: no foreign accent, 9: heavy foreign accent). Each participant rated 10 utterances per speaker per system. Participants were also asked to rate utterances from L1 and L2 speakers, which served as a reference. Results are shown in Table 3. Both systems obtained significantly lower ratings (i.e., more native-accented) than the original L2 utterances ($p \ll 0.001$). The baseline system reduced accentedness by 28.9%, whereas the proposed system achieved a much greater reduction (67.0%); this difference was statistically significant ($p \ll 0.001$).

5.2. Acoustic quality test

Participants were asked to rate the acoustic quality of an utterance using a standard 5-point scale mean opinion score (MOS). The ratings scale and their corresponding speech quality and level of distortion labels are listed in Table 2. Listeners were provided with reference audio samples with different MOS ratings so they could calibrate themselves [35]. These reference samples were taken from the 2018 Voice Conversion Challenge dataset [36]. Each listener rated 10 utterances per speaker per system. Listeners also rated utterances from L1 and L2 speakers, as in the accentedness test. Results are shown in Table 3. It can be observed that both baseline and proposed systems obtained similar MOS, and the differences were not statistically significant ($p = 0.91$). The original L1 and L2 utterances received the highest MOS ratings, as expected.

5.3. Voice identity test

To evaluate voice identity, we conducted an ABX test where participants were presented with two audio samples, one from the L1 speaker, and the other from the L2 speaker, followed by the accent-converted sample. Then, participants had to decide which audio sample (L1 or L2) the synthesized speech was most similar to, and then rate the confidence in their decision using a 7-point scale (7: extremely confident; 5: quite a bit confident; 3: somewhat confident; 1: not confident at all). Following Felps *et al.* [37], the decision and confidence level were then collapsed to form a 14-point VSS (Voice Similarity Score) scale: -7 (definitely L1) to +7 (definitely L2). To minimize the effect of accent, all audio samples were played in reverse and had different linguistic content. Each listener rated 15 of such combinations per speaker per system. The order in which L1 and L2 utterances appeared were randomized. Results are shown in Table 4. Across both systems, more than 80% of participants were "quite a bit" confident that the accent-converted speech had the same speaker identity as the original L2 speaker. This indicates that both systems were able to retain the speaker identity, though the baseline model performs marginally better ($p = 0.03$). This result is remarkable considering that the baseline system builds a dedicated one-to-one model, whereas the proposed system is a many-to-many system that is tested on unseen speakers (zero-shot).

Table 4: *Voice similarity ratings for the two FAC systems.*

	Baseline	Proposed
Prefer L2 speaker	84.33%	82.00%
Average rater confidence	5.59	5.30

6. Discussion

We have presented an accent-conversion system that can transform utterances from unseen L2 speakers (i.e., zero-shot) to sound as if they were produced with a native accent, and compared it against a state-of-the-art baseline that requires building a dedicated system for each L2 speaker. The proposed system[1] achieves near-native ratings of accentedness, significantly outperforming those achieved by the baseline system. This is made possible by training a model to "translate" L2 bottleneck features into equivalent L1 bottleneck features, in this way correcting the pronunciation errors in the L2 utterance. The translator is trained using multiple speakers from six different accents, so it learns to handle many more mispronunciation variations than it would from a one-to-one system. This might explain why we observe such a significant drop in accentedness ratings compared to the baseline system. Further, our proposed system achieves similar ratings of MOS and speaker identity as the baseline system, despite the fact that our system was evaluated on unseen speakers whereas the baseline system only works on seen L2 speakers.

Synthesized speech for both systems received lower MOS ratings than the original utterances. This is likely due to the fact that both systems use an independently-trained vocoder. Thus, a future direction to improve MOS ratings is to jointly train the synthesizer and the vocoder. We restricted our experiments to perform FAC on accents that were already present in the training set. Future work is necessary to test the system's ability to handle unseen accents at the input. Another future research direction is to make the model robust when synthesizing long utterances. The current architecture uses a location-sensitive attention mechanism, which is prone to fail with long utterances. Thus, alternative attention mechanism like the gaussian mixture attention [38, 39] could be used instead. Finally, transformer networks [40] can also be used as an alternative for the seq2seq models. Transformer networks could also help reducing synthesis time through non-autoregressive sequence generation [41] and perform real-time accent conversion.

7. Acknowledgments

This work was supported by NSF awards 1619212 and 1623750.

[1] Audio samples from the two sytems can be found at: https://warisqr007.github.io/demos/zero-shot-reference-free-ac/

8. References

[1] D. Felps, H. Bortfeld, and R. Gutierrez-Osuna, "Foreign accent conversion in computer assisted pronunciation training," *Speech communication*, vol. 51, no. 10, pp. 920–932, 2009.

[2] S. Ding, C. Liberatore, S. Sonsaat, I. Lučić, A. Silpachai, G. Zhao, E. Chukharev-Hudilainen, J. Levis, and R. Gutierrez-Osuna, "Golden speaker builder–An interactive tool for pronunciation training," *Speech Communication*, vol. 115, pp. 51–66, 2019.

[3] L. Sun, H. Wang, S. Kang, K. Li, and H. M. Meng, "Personalized, Cross-Lingual TTS Using Phonetic Posteriorgrams," in *INTERSPEECH*, 2016, pp. 322–326.

[4] Y. Oshima, S. Takamichi, T. Toda, G. Neubig, S. Sakti, and S. Nakamura, "Non-native speech synthesis preserving speaker individuality based on partial correction of prosodic and phonetic characteristics," in *Proc. Interspeech 2015*, 2015, pp. 299–303.

[5] O. Turk and L. M. Arslan, "Subband based voice conversion," in *Seventh International Conference on Spoken Language Processing*, 2002.

[6] F. Biadsy, R. J. Weiss, P. J. Moreno, D. Kanevsky, and Y. Jia, "Parrotron: An end-to-end speech-to-speech conversion model and its applications to hearing-impaired speech and speech separation," *arXiv preprint arXiv:1904.04169*, 2019.

[7] G. Zhao and R. Gutierrez-Osuna, "Using phonetic posteriorgram based frame pairing for segmental accent conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 10, pp. 1649–1660, 2019.

[8] G. Zhao, S. Ding, and R. Gutierrez-Osuna, "Converting Foreign Accent Speech Without a Reference," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 2367–2381, 2021.

[9] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *CVPR*. IEEE, 2009, pp. 951–958.

[10] F.-L. Xie, F. K. Soong, and H. Li, "A KL divergence and DNN-based approach to voice conversion without parallel training sentences," in *Interspeech*, 2016, pp. 287–291.

[11] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, and R. Skerrv-Ryan, "Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions," in *ICASSP*. IEEE, 2018, pp. 4779–4783.

[12] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," in *ICML*. PMLR, 2018, pp. 2410–2419.

[13] S. Aryal and R. Gutierrez-Osuna, "Reduction of non-native accents through statistical parametric articulatory synthesis," *The Journal of the Acoustical Society of America*, vol. 137, no. 1, pp. 433–446, 2015.

[14] D. Felps, C. Geng, and R. Gutierrez-Osuna, "Foreign accent conversion through concatenative synthesis in the articulatory domain," *IEEE TASLP*, vol. 20, no. 8, pp. 2301–2312, 2012.

[15] S. Aryal and R. Gutierrez-Osuna, "Data driven articulatory synthesis with deep neural networks," *Computer Speech & Language*, vol. 36, pp. 260–273, 2016.

[16] S. Aryal and R. Gutierrez-Osuna, "Can voice conversion be used to reduce non-native accents?" in *ICASSP*. IEEE, 2014, pp. 7879–7883.

[17] H. Miyoshi, Y. Saito, S. Takamichi, and H. Saruwatari, "Voice Conversion Using Sequence-to-Sequence Learning of Context Posterior Probabilities and Evaluation of Dual Learning," *IEICE Technical Report; IEICE Tech. Rep.*, vol. 117, no. 160, pp. 9–14, 2017.

[18] J.-X. Zhang, Z.-H. Ling, and L.-R. Dai, "Non-parallel sequence-to-sequence voice conversion with disentangled linguistic and speaker representations," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 540–552, 2019.

[19] S. Liu, D. Wang, Y. Cao, L. Sun, X. Wu, S. Kang, Z. Wu, X. Liu, D. Su, and D. Yu, "End-to-end accent conversion without using native utterances," in *ICASSP*. IEEE, 2020, pp. 6289–6293.

[20] G. Zhao, S. Ding, and R. Gutierrez-Osuna, "Foreign Accent Conversion by Synthesizing Speech from Phonetic Posteriorgrams," in *INTERSPEECH*, 2019, pp. 2843–2847.

[21] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, and S. Bengio, "Tacotron: Towards end-to-end speech synthesis," *arXiv preprint arXiv:1703.10135*, 2017.

[22] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. Lopez Moreno, and Y. Wu, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," *Advances in neural information processing systems*, vol. 31, 2018.

[23] S. Liu, J. Zhong, L. Sun, X. Wu, X. Liu, and H. Meng, "Voice Conversion Across Arbitrary Speakers Based on a Single Target-Speaker Utterance," in *Interspeech*, 2018, pp. 496–500.

[24] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "ACVAE-VC: Non-parallel voice conversion with auxiliary classifier variational autoencoder," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 9, pp. 1432–1443, 2019.

[25] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[26] S. Ding, G. Zhao, and R. Gutierrez-Osuna, "Accentron: Foreign accent conversion to arbitrary non-native speakers using zero-shot learning," *Computer Speech Language*, vol. 72, p. 101302, 2022.

[27] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *ICASSP*. IEEE, 2018, pp. 4879–4883.

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

[29] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, and P. Schwarz, "The Kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011.

[30] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[31] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[32] S. Weinberger, "Speech accent archive," *George Mason University*, 2015.

[33] J. Kominek and A. W. Black, "The CMU Arctic speech databases," in *Fifth ISCA workshop on speech synthesis*, 2004, pp. 223–224.

[34] G. Zhao, S. Sonsaat, A. Silpachai, I. Lucic, E. Chukharev-Hudilainen, J. Levis, and R. Gutierrez-Osuna, "L2-ARCTIC: A non-native English speech corpus," in *INTERSPEECH*, 2018, pp. 2783–2787.

[35] P. C. Loizou, *Speech quality assessment*. Springer, 2011, pp. 623–654.

[36] J. Lorenzo-Trueba, J. Yamagishi, T. Toda, D. Saito, F. Villavicencio, T. Kinnunen, and Z. Ling, "The voice conversion challenge 2018: Database and results," 2018.

[37] D. Felps and R. Gutierrez-Osuna, "Developing objective measures of foreign-accent conversion," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 1030–1040, 2010.

[38] R. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. Weiss, R. Clark, and R. A. Saurous, "Towards end-to-end prosody transfer for expressive speech synthesis with tacotron," in *ICML*. PMLR, 2018, pp. 4693–4702.

[39] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:1308.0850*, 2013.

[40] W.-C. Huang, T. Hayashi, Y.-C. Wu, H. Kameoka, and T. Toda, "Voice transformer network: Sequence-to-sequence voice conversion using transformer with text-to-speech pretraining," *arXiv preprint arXiv:1912.06813*, 2019.

[41] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech 2: Fast and high-quality end-to-end text to speech," *arXiv preprint arXiv:2006.04558*, 2020.