Converting Foreign Accent Speech Without a Reference

Guanlong Zhao †, Shaojin Ding ‡, and Ricardo Gutierrez-Osuna, Senior Member, IEEE

Abstract—Foreign accent conversion (FAC) is the problem of generating a synthetic voice that has the voice identity of a second-language (L2) learner and the pronunciation patterns of a native (L1) speaker. This synthetic voice has been referred to as a “golden-speaker” in the pronunciation-training literature. FAC is generally achieved by building a voice-conversion model that maps utterances from a source (L1) speaker onto the target (L2) speaker. As such, FAC requires that a reference utterance from the L1 speaker be available at synthesis time. This greatly restricts the application scope of the FAC system. In this work, we propose a “reference-free” FAC system that eliminates the need for reference L1 utterances at synthesis time, and transforms L2 utterances directly. The system is trained in two steps. First, a conventional FAC procedure is used to create a golden-speaker using utterances from a reference L1 speaker (which are then discarded) and the L2 speaker. Second, a pronunciation-correction model is trained to convert L2 utterances to match the golden-speaker utterances obtained in the first step. At synthesis time, the pronunciation-correction model directly transforms a novel L2 utterance into its golden-speaker counterpart. Our results show that the system reduces foreign accents in novel L2 utterances, achieving a 20.5% relative reduction in word-error-rate of an American English automatic speech recognizer and a 19% reduction in perceptual ratings of foreign accentness obtained through listening tests. Over 73% of the listeners also rated golden-speaker utterances as having the same voice identity as the original L2 utterances.

Index Terms—Accent conversion, acoustic model, sequence-to-sequence voice conversion, speech modification, speech synthesis.

I. INTRODUCTION

FOREIGN accent conversion (FAC) [1] aims to create a synthetic voice that has the voice identity (or timbre) of a non-native speaker but the pronunciation patterns (or accent) of a native speaker. In the context of computer-assisted pronunciation training [1]–[4], this synthetic voice is often referred to as a “golden speaker” for the non-native speaker –a second-language (L2) learner. The rationale is that the golden speaker is a better target for the L2 learner to imitate than an arbitrary native speaker, because the only difference between the golden speaker and the L2 learner’s own voice is the accent, which makes mispronunciations more salient. In addition to pronunciation training, FAC finds applications in movie dubbing [5], personalized Text-To-Speech (TTS) synthesis [6], [7], and improving automatic speech recognition (ASR) performance [8].

The main challenge in FAC is that one does not have ground-truth data for the desired golden speaker, since, in general, the L2 learner is unable to produce speech with a native accent. Therefore, it is not feasible to apply conventional voice-conversion techniques to the FAC problem. Previous solutions work around this issue by requiring a reference utterance from a native (L1) speaker at synthesis time. But this limits the types of pronunciation practice that FAC techniques can provide, e.g., the L2 learner can only practice sentences that have already been prerecorded by the reference L1 speaker.

To address this issue, we propose a new FAC system that does not require a reference L1 utterance at inference time. We refer to this type of FAC system as reference-free. Assume that we have a training set of parallel utterances from the L2 learner and from a reference L1 speaker. The training pipeline consists of two steps. In step one, we build an L2 speech synthesizer [9] that maps speech embeddings (see below) from L2 utterances into their corresponding Mel-spectrograms. The speech embeddings are extracted using an acoustic model trained on a large corpus of native speech, so they are speaker-independent [10], [11]. We then drive the L2 synthesizer with speech embeddings extracted from the L1 utterances. This results in a set of golden-speaker utterances that have the voice identity of the L2 learner (since they are generated from the L2 synthesizer) and the pronunciation patterns of the L1 speaker (since the input is obtained from an L1 utterance). The L1 utterances can be discarded at this point. In the second (and key) step, we train a pronunciation-correction model that converts the
L2 utterances to match the golden-speaker utterances obtained in
the first step, which serves as a target. During inference time, we
can then feed a new L2 utterance to the pronunciation-correction
model, which then generates its “accent free” counterpart.

The pronunciation-correction model is based on a
state-of-the-art sequence-to-sequence (seq2seq) voice
conversion framework proposed by Zhang et al. [12], which we
use as a baseline. Their system consists of an encoder to extract
hidden representations of the input features (e.g., Mel-spectra),
an attention mechanism to learn the alignment between the
input and output sequences, a decoder to predict the output
Mel-spectrograms, and multi-task phoneme classifiers to help
stabilize the training process. During our internal evaluation of
the baseline system, we found that it had difficulty converting
between an L2 and an L1 speaker because L2 utterances tend to
have a significant amount of disfluency and hesitations, which
makes it hard for the attention mechanism to properly align
input and output sequences. To address this issue, our system
includes a forward-and-backward decoding technique [13],
[14] in the pronunciation-correction model to help the attention
mechanism and decoder to fully utilize the information in the
input data. The rationale is that, by forcing the decoder to
compute the attention alignments from both the past and
backward directions during training, we can make the decoder
incorporate useful contextual information from both the past
and future when producing the alignment. Throughout this
study, we use a high-quality WaveGlow [15] real-time neural
vocoder to convert Mel-spectrograms to speech waveform.

The manuscript is organized as follows. Section II reviews
prior approaches on FAC as well as related work in seq2seq
voice conversion. Section III describes the proposed reference-
free FAC system. Sections V, IV, and VI present the objective
and subjective evaluation results and an in-depth discussion of
these results. Lastly, we summarize the findings of this work in
Section VII and point out future research directions. We include
three appendices that provide related details.

II. RELATED WORK

A. Conventional FAC Methods

FAC is related to the more general problem of voice conver-
sion (VC) [16]. In VC, one seeks to transform a source speaker’s
speech into that of a (known) target speaker. The conversion
aims to match the voice characteristics of the target speaker,
which include vocal tract configurations, glottal characteristics,
pitch range, pronunciation, and speaking rate; ideally, the only
information retained from the source speech is its linguistic
content, i.e., the words that were uttered. In contrast with VC,
FAC seeks to combine the linguistic content and pronunciation
characteristics of the source speaker with the voice identity of the
target speaker. This is a more challenging problem than VC for
two reasons. First, FAC lacks ground-truth since generally there
are no recordings of the L2 speaker producing speech with the
desired native target accent. But, more importantly, FAC requires
decomposing the speech into voice identity and accent, whereas
VC does not. Several techniques have been proposed to perform
this decomposition, which can be grouped into articulatory and
acoustic methods. The basic strategy in articulatory methods
is to build an articulatory synthesizer for the L2 speaker, that is,
a mapping from the speaker’s articulatory trajectories (e.g.,
tongue and lip movements) to his or her acoustics features (e.g.,
Mel cepstra). Once complete, the L2 speaker’s articulatory
synthesizer is driven by articulatory trajectories from an L1
speaker to produce “accent-free” speech.2 A number of tech-
niques can be used to build the articulatory synthesizer, including
unit-selection [18], GMMs [19], and DNNs [20].

Decoupling voice identity from accent in the articulatory
domain is intuitive, but impractical in most cases since col-
lecting articulatory data is expensive and requires specialized
equipment.3 In contrast, decoupling voice identity from accent
in the acoustic domain is more practical since it only requires
recording speech with a microphone, but is more challenging
from a speech-processing standpoint. The conventional ap-
proach used in VC (pairing source and target frames via dynamic
time warping; DTW) cannot be used in FAC, since it would result
in a model that maps native-accented source into non-native-
accented target speech. Instead, source and target frames have
to be paired based on their linguistic similarity. In early work,
Aryal and Gutierrez-Osuna [24] replaced DTW with a technique
that matched source (L1) and target (L2) frames based on their
MFCC similarity after performing vocal tract length (VTL)
normalization. Then, they trained a GMM with those frame pairs to
map source L1 utterances to have the target L2 speaker’s identity,
while retaining the native pronunciations. More recently, Zhao
et al. [25] used a speaker-independent acoustic model (i.e., from
an ASR system) to estimate the posterior probability that each
frame belonged to a set of pre-defined phonetic units—a phonetic
posteriorgram (PPG) [26]. Once a PPG had been computed for
each source and target frame in the corpus, the two were paired
in a many-to-many fashion based on the similarity between
their respective PPGs [11], [25]. In their study, matching source
and target frames based on their PPG similarity achieved better
ratings on accentedness and acoustic quality than matching them
based on the VTL-normalized MFCC similarity of Aryal and
Gutierrez-Osuna [24].

B. FAC Methods Using Sequence-to-Sequence Models

More recently, Zhao et al. [27] have used sequence-to-
sequence (seq2seq) models to perform FAC. In their approach,
a seq2seq speech synthesizer is trained to convert PPGs to
Mel-spectra using recordings from the L2 speaker. Then, golden-
speaker utterances are generated by driving the seq2seq syn-
thesizer with PPGs extracted from an L1 utterance, a process
that reminisces articulatory-based methods (i.e., if PPGs are
viewed as articulatory information). Their method produced
speech that was significantly less accented than the original
L2 speech. Seq2seq models have also garnered much attention
in the VC literature since, unlike prior frame-by-frame VC

2This process can be likened to “voice puppetry” [17] where the puppet is the
articulatory synthesizer and the strings are the native speaker’s articulations.

3Articulatory measurements can be performed via electromagnetic articu-
lography [18] ultrasound imaging [21], palatography [22], and more recently,
real-time MRI [23].
models [28]–[33], they can convert segmental and prosody features simultaneously, leading to better conversion performance. Miyoshi et al. [34] built a seq2seq model that mapped source context posterior probabilities to the target’s; they obtained better speech individuality ratings (but worse audio quality) than a baseline without the context posterior mapping process. Zhang et al. [35] concatenated bottleneck features and Mel-spectrograms from a source speaker, used a seq2seq model to convert the concatenated source features into the target Mel-spectrogram, and finally recovered the speech waveform with a WaveNet [36] vocoder. This model outperformed the best-performing system from the 2018 Voice Conversion Challenge [37]. Zhang et al. then applied text supervision [12] on top of [35] to resolve some of the mispronunciations and artifacts in the converted speech. More recently, they extended their framework to the non-parallel condition [38] with trainable linguistic and speaker embeddings. Other notable sequence-to-sequence VC works include [39], which proposed a novel loss term that enforced attention weight diagonality to stabilize the seq2seq training; the Parrottron [8] system, which used large-scale corpora and seq2seq models to normalize arbitrary speaker voices to a synthetic TTS voice; and [40], which used a fully convolutional seq2seq model instead of conventional recurrent neural networks (RNNs, e.g., LSTM) because RNNs are costly to train and difficult to optimize for parallel computing.

C. Prior Reference-Free FAC Approach

To the best of our knowledge, the only prior work on reference-free FAC is a recent study by Liu et al. [41]. Their system used a speaker encoder, a multi-speaker TTS model, and an ASR encoder. The speaker encoder and the TTS model are trained with L1 speech only, and the ASR encoder is trained on speech data from L1 speakers and the target L2 speaker. During testing, they use the speaker encoder and ASR encoder to extract speaker embeddings and linguistic representations from the input L2 testing utterance, respectively. Then, they concatenate the two and feed them to the multi-speaker TTS model, which then generates the accent-converted utterance. Their evaluations suggested that the converted speech had a near-native accent, but did not capture the voice identity of the target L2 speaker because it had to be interpolated by their multi-speaker TTS. Our proposed method avoids this problem since our pronunciation-correction model as L2-GS utterances since they are obtained directly from L2 utterances (i.e., in a reference-free fashion). Critical in this process is the generation of the speaker embeddings, which we describe first.

A. Extracting Speaker-Independent Speech Embeddings

We use an acoustic model (AM) to generate a speaker-independent (SI) speech embedding for an input (L1 or L2) utterance. Our AM is a Factorized Time Delayed Neural Network (TDNN-F) [42], [43], a feedforward neural network that utilizes time-delayed input in its hidden layers to model long term temporal dependencies. TDNN-F can achieve performance on Large Vocabulary Continuous Speech Recognition (LVCSR) tasks that is comparable to that of AMs based on recurrent structures (e.g., Bi-LSTMs), but is more efficient during training and inference due to its feedforward nature [42]. To produce an SI speech embedding, we concatenate each acoustic feature vector (40-dim MFCC) with an i-vector (100-dim) of the corresponding speaker [44] and use them as inputs to the AM, which we then train on a large corpus from a few thousand native speakers (Librispeech [45]).

4 The AM is trained following the Kaldi [46] “tdnn_1 d” configuration of the TDNN-F model. We use the full training set (960 hours) in the Librispeech

III. METHOD

Our proposed approach to reference-free FAC is illustrated in Figure 1. The system requires a parallel corpus of utterances from the L2 speaker and a reference L1 speaker. As outlined in the introduction and shown in the figure, the training process consists of two steps. In a first step, we build a speech synthesizer for the L2 speaker that converts speech embeddings into Mel-spectrograms. We then drive the L2 synthesizer with a set of utterances from the reference L1 speaker, to produce a set of golden-speaker utterances (i.e., L2 voice identity with L1 pronunciation patterns). We refer to these as L1 golden-speaker (L1-GS) utterances, since they are obtained using L1 utterances as a reference. The L1 utterances can be discarded at this point. In a second step, we build a pronunciation-correction model that directly transforms L2 utterances to match their corresponding L1-GS utterances obtained earlier. Once the pronunciation correction model is trained, in the testing stage, a new L2 utterance is processed by the pronunciation-correction model to create its “accent-free” counterpart (L2-GS).

Fig. 1. Overall workflow of the proposed system. L1: native; L2: non-native; GS: golden speaker; SI: speaker independent. The training stage consists of two steps. In step 1, we use a conventional FAC procedure to generate a set of golden-speaker utterances (L1-GS), which serve as targets for step 2. In step 2, we train a pronunciation-correction model that converts L2 utterances into the L1-GS utterances obtained earlier. Once the pronunciation correction model is trained, in the testing stage, a new L2 utterance is processed by the pronunciation-correction model to create its “accent-free” counterpart (L2-GS).
As part of this study, we evaluated three different speech embeddings:

- **Senone phonetic posteriorgram (Senone-PPG):** The output from the final softmax layer of the AM, which is high dimensional (6024 senones) and contains fine-grained information about the pronunciation pattern in the input utterance.
- **Bottleneck feature (BNF):** The output of the layer prior to the final softmax layer of the AM. The BNF contains rich classifiable information for a phoneme recognition task, but lower dimensionality (256).
- **Monophone phonetic posteriorgram (Mono-PPG):** The phonetic posteriorgram obtained by collapsing the senones into monophone symbols (346 monophones with word positions, e.g., word-initials, word-finals). For each monophone symbol, we aggregate the probability mass of all the senones that share the same root monophone. Figure 2 visualizes the Mono-PPG of a spoken word. We omit the visualization of the other two speech embeddings since they are more difficult to interpret.

### B. Step 1: Generating a Reference-Based Golden-Speaker (LI-GS)

The speech synthesizer is based on a modified Tacotron2 architecture\(^5\) [9], and is illustrated in Figure 3. The model follows a general encoder-decoder (or seq2seq) paradigm with an attention mechanism. Conceptually, an encoder-decoder architecture uses an encoder (usually a recurrent neural network; RNN) to “consume” input sequences and generate a high-level hidden representation sequence. Then, a decoder (an RNN with an attention mechanism) processes the hidden representation sequence. Finally, we pass the hidden linguistic representation sequence to the decoder, which consists of a location-sensitive attention mechanism [47] and a decoder LSTM, to predict the raw Mel-spectrogram. We note that the input and output sequences of the speech synthesizer have the same length,\(^6\) and thus, the speech synthesizer only models the speaker identity and retains the phonetic and prosodic cues carried by the input speech embeddings.

Formally, let \([a; b]\) represent the operation of concatenating vectors \(a\) and \(b\), \(h = [h_1, \ldots, h_T]\) be the full sequence of hidden linguistic representation from the encoder and \((\cdot)^\top\) denote the matrix transpose. At the \(i\)-th decoding time step, applying the location-sensitive attention mechanism, the attention context vector \(c_i\) is the weighted sum of \(h\),

\[
c_i = \alpha_i \cdot h^\top,
\]

\[
\alpha_i = \text{AttentionLayers} \left( q_i, \alpha_{i-1}, h \right) = [\alpha_i^1, \ldots, \alpha_i^n^T],
\]

\[
q_i = \text{AttentionLSTM} \left( q_{i-1}; [c_{i-1}; \text{DecoderPreNet} \left( \hat{y}_{i-1}^{mel} \right)] \right),
\]

\[
\alpha_i^j = \frac{\exp(e_{ij})}{\sum_{j=1}^n \exp(e_{ij})},
\]

\[
e_{ij} = v^\top \tanh \left( W q_i + V h_j + U f_i^j + b \right),
\]

\[
f_i = F \ast \alpha_{i-1} = [f_i^1, \ldots, f_i^n], F \in \mathbb{R}^{k \times r}.
\]

\(\alpha_i = [\alpha_i^1, \ldots, \alpha_i^n^T]\) are the attention weights, \(q_i\) is the output of the attention LSTM, and \(\hat{y}_{i-1}^{mel}\) is the predicted raw Mel-spectrogram from the previous time step. \(v, W, V, U, b, F\) are learnable parameters of the attention layers. \(F\) contains \(k\)-1 learnable kernels with kernel size \(r\), and \(f_i^j \in \mathbb{R}^k\) is the result of convolving \(\alpha_{i-1}\) at position \(j\) with \(F\).

Next, let \(d_i\) be the output of the decoder LSTM at decoding time step \(i\), and \(\hat{y}_{i}^{mel}\) be the new raw Mel-spectrum prediction, we have,

\[
d_i = \text{DecoderLSTM} \left( d_{i-1}, [q_i; c_i] \right),
\]

\(a\) weighted sum of the hidden representation sequence) to summarize the contextual information. The decoder RNN reads the attention context vectors and predicts the output sequence in an autoregressive manner.

Our speech synthesizer takes the speech embeddings as input. Then, if the input speech embeddings have high dimensionality (e.g., Senone-PPGs), we reduce their dimensions through a learnable input PreNet. This step is essential for the model to converge when using high-dimensional speech embeddings as input. For speech embeddings with lower dimensionality, such as Mono-PPGs and BNFs, we skip the input PreNet. The speech embeddings are then passed through multiple 1-D convolutional layers, which model longer-term context. Next, an encoder (one Bi-LSTM) converts the convolutions into a hidden linguistic representation sequence. Finally, we pass the hidden linguistic representation sequence to the decoder, which consists of a location-sensitive attention mechanism [47] and a decoder LSTM, to predict the raw Mel-spectrogram. We note that the input and output sequences of the speech synthesizer have the same length,\(^6\) and thus, the speech synthesizer only models the speaker identity and retains the phonetic and prosodic cues carried by the input speech embeddings.

\(^5\)To facilitate the method description and maintain consistency with prior literature, we adopt the following terminologies from Tacotron2: PreNet: Two fully connected layers with a ReLU nonlinearity; PostNet: Five stacked 1-D convolutional layers; LinearProjection: One fully connected layer.

\(^6\)A recent study [48] (published while this manuscript was under review) used a conversion model similar to the one used in our work. The authors observed that if the temporal structure (such as the length) of the input and output sequences were the same, then removing the attention module did not hurt performance, which suggests a potential path to further simplify the model structure of the speech synthesizer we used here.
\[
\hat{y}^\text{mel}_i = \text{LinearProjection}^\text{mel}(\{d_i; c_i\}).
\] (8)

At each time step, to determine if the decoder prediction reaches the end of an utterance, we compute a binary stop token (1: stop; 0: continue) using a separate trainable fully connected layer,

\[
\hat{y}_i^\text{stop} = \begin{cases} 
1 & \text{Sigmoid}(\text{LinearProjection}^\text{stop}(\{d_i; c_i\})) \geq 0.5 \\
0 & \text{Sigmoid}(\text{LinearProjection}^\text{stop}(\{d_i; c_i\})) < 0.5 
\end{cases}
\] (9)

The original Tacotron 2 was designed to accept character sequences as input, which are significantly shorter than our speech embedding sequences. For example, each sentence in our corpus contains 41 characters on average, whereas the corresponding speech embedding sequence has a few hundred frames. Therefore, the vanilla location-sensitive attention mechanism might fail, as pointed out in [35]. As a result, the inference would be ill-conditioned and would generate non-intelligible speech. Following a preliminary study [27] of this work, we add locality constraint to the attention mechanism. Speech signals have a strong temporal-continuity and progressive nature. To capture the phonetic context, we only need to look at the speech embeddings in a small local window. Inspired by this, at each decoding step during training, we constrain the attention mechanism to only consider the hidden linguistic representation within a fixed window centered on the current frame, i.e., let,

\[
h = [h_{i-w}, \ldots, h_i, \ldots, h_{i+w}, 0, \ldots, 0],
\] (10)

where \(w\) is the window size. Consequentially, we replace eq. (2) with eq. (11),

\[
\alpha_i = \text{AttentionLayers}\left(q_i, \alpha_{i-1}, \hat{h}\right).
\] (11)

Finally, to further improve the synthesis quality, the speech synthesizer appends a PostNet after the decoder to predict residual spectral details from the raw Mel-spectrum prediction, and then adds the spectral residuals to the raw Mel-spectrum,

\[
\hat{y}_i^\text{PostNet} = \hat{y}^\text{mel}_i + \text{PostNet}(\hat{y}^\text{mel}_i).
\] (12)

The advantage of the PostNet is that it can see the entire decoded sequence. Therefore, the PostNet can use both past and future information to correct the prediction error for each individual frame [49].

The loss function for training this speech synthesizer is,

\[
L = w_1 \left( \|Y_{\text{mel}} - \hat{Y}_{\text{mel}}^\text{Decoder} \|_2 + \|Y_{\text{mel}} - \hat{Y}_{\text{mel}}^\text{PostNet} \|_2 \right) + w_2 \text{CE}(Y_{\text{stop}}, \hat{Y}_{\text{stop}}),
\] (13)

where \(Y_{\text{mel}}\) is the ground-truth Mel-spectrogram; \(\hat{Y}_{\text{mel}}^{\text{Decoder}}\) and \(\hat{Y}_{\text{mel}}^{\text{PostNet}}\) are the predicted Mel-spectrograms from the decoder and PostNet, respectively; \(Y_{\text{stop}}\) and \(\hat{Y}_{\text{stop}}\) are the ground-truth and predicted stop token sequences; \(\text{CE}(\cdot)\) is the cross-entropy loss; \(w_1\) and \(w_2\) control the relative importance of each loss term.

The predicted Mel-spectrograms are converted back to audio waveforms using a WaveGlow neural vocoder trained on the L2 utterances (cf. Section III-D for more details). We then drive the L2 synthesizer with a set of utterances from the reference L1 speaker, to produce the L1-GS utterances that are used in Step 2.
Specifically, let $x_i$ be the $i$-th feature vector in the sequence, the input $X = [x_1, \ldots, x_{T_{in}}]$ to the conversion system is the concatenation of the bottleneck features (i.e., BNFs, cf. Section III-A) and Mel-spectrogram computed from the L2 utterance. The output sequence is denoted by $Y_{mel} = [y_1^{mel}, \ldots, y_{T_{out}}^{mel}]$ where $y_i^{mel}$ is the $i$-th Mel-spectrum of the L1-GS utterance. A two-layer Pyramid-Bi-LSTM encoder [50] with a down-sampling rate of two consumes the input sequence and produces the encoder hidden embeddings $h = [h_1, \ldots, h_{i/2}], \ldots, h_{(T_{in}/2)}$, where $h_{i/2}$ is one encoder hidden embedding vector, and $[\cdot]$ is the floor-rounding operator. The first Bi-LSTM layer does the recurrent computations on $X$ and outputs $h_{layer1} = [h_1^{layer1}, \ldots, h_{T_{layer1}}^{layer1}]$. We then concatenate each two of the consecutive frames in $h_{layer1}$ to form $[[h_{layer1}^{1}, h_{layer1}^{2}], \ldots, [h_{T_{layer1} - 1}^{layer1}, h_{T_{layer1}}^{layer1}]]$. Finally, we feed the concatenated vectors to the second Bi-LSTM layer to produce $h$. In the case that we have an odd number of frames in the input sequence, we drop the last frame, which is generally a silent frame. The down-sampling effectively reduces the sequence length of the input, which speeds up the encoder computation by a factor of two and makes it easier for the attention mechanism to learn a meaningful alignment between the input and output sequences.

The decoder in this model has a similar neural-network structure as the speech synthesizer decoder in Section III-B (Figure 3), with only two differences: (1) to replicate Zhang et al. [12], we use the forward-attention technique [51] instead of eq. (4) to normalize the attention weights; (2) the locality constraint defined in equations (10) and (11) is discarded. The decoder predicts the output raw Mel-spectrogram sequence $\tilde{Y}_{mel}^{Decoder} = [\tilde{y}_1^{mel}, \ldots, \tilde{y}_{T_{out}}^{mel}]$ and the stop token sequence $\tilde{Y}_{stop} = [\tilde{y}_{stop}, \ldots, \tilde{y}_{T_{stop}}^{mel}]$, following equations (8) and (9), respectively. $\tilde{Y}_{mel}^{Decoder}$ is also processed through a PostNet to generate a residual-compensated Mel spectrogram $\tilde{Y}_{mel}^{PostNet}$, following eq. (12). As in the previous step, $\tilde{Y}_{mel}^{PostNet}$ is converted back to audio waveforms using a WaveGlow neural vocoder trained on the L2 utterances.

In addition, the baseline system uses multi-task learning [52], [53] to make the synthesized pronunciations more stable. Two independent phoneme classifiers, each containing one fully-connected layer and a softmax operation, are added to predict the input and output phoneme sequences $\hat{Y}_{in}^{P} = [\hat{y}_1^{inP}, \ldots, \hat{y}_{T_{in}}^{inP}]$ and $\hat{Y}_{out}^{P} = [\hat{y}_1^{outP}, \ldots, \hat{y}_{T_{out}}^{outP}]$, respectively. These phoneme classifiers are only used during training and are discarded in inference. $c_i$ and $q_i$ are defined in the same manner as in equations (1) and (3).

$$\hat{y}_i^{inP} = \text{PhonemeClassifier}_{inP}(h_i).$$  \hspace{1cm} (14)

$$\hat{y}_i^{outP} = \text{PhonemeClassifier}_{outP}([q_i; c_i]).$$  \hspace{1cm} (15)

---

7 Zhang et al. [12] use BNFs in their implementation, and we follow this design choice to replicate their system.
The final loss function of the baseline system becomes,

\[ L_{\text{base}} = w_1 \left( \| Y_{\text{mel}} - \hat{Y}_{\text{mel}}^{\text{Decoder}} \|_2 + \| Y_{\text{mel}} - \hat{Y}_{\text{mel}}^{\text{PostNet}} \|_2 \right) \\
+ w_2 \text{CE} \left( Y_{\text{stop}}, \hat{Y}_{\text{stop}} \right) \tag{16} \]

\[ + w_3 \left( \text{CE} \left( Y_{\text{inP}}, \hat{Y}_{\text{inP}} \right) + \text{CE} \left( Y_{\text{outP}}, \hat{Y}_{\text{outP}} \right) \right), \]

where \( Y_{\text{inP}}, Y_{\text{outP}} \) are the ground-truth input and output phoneme sequence, respectively.

To improve predictive performance, we propose a modification to the baseline system that applies forward-and-backward decoding during the training process. The forward-and-backward decoding technique maintains two separate decoders, i.e., the forward and backward decoders. The forward decoder processes the encoder outputs in the forward direction, whereas the backward decoder reads the encoder outputs in reverse direction. Different variations of this technique have been applied to TTS [14] and ASR [13]. Figure 5 shows an overview of this procedure. During training, we add a backward decoder to the baseline model. The backward decoder has the same structure as the existing decoder (denoted as the forward decoder) but with a different set of weights. The backward decoder functions the same as the forward decoder except that it processes the encoder’s output in reverse order and predicts the output Mel-spectrogram \( \hat{Y}_{\text{mel}}^{\text{bwd}} \) reversely as well. The backward decoder, like its forward counterpart, also predicts its own set of stop tokens \( T_{\text{stop}} \), output phoneme labels \( Y_{\text{outP}}^{\text{bwd}} \), and uses the shared PostNet to predict a refined Mel-spectrogram \( \hat{Y}_{\text{mel}}^{\text{bwd} \text{PostNet}} \). The loss terms contributed by adding this backward decoder are,

\[ L_{\text{bwd}} = w_1 \left( \| Y_{\text{mel}} - \hat{Y}_{\text{mel}}^{\text{bwd}} \|_2 + \| Y_{\text{mel}} - \hat{Y}_{\text{mel}}^{\text{bwd} \text{PostNet}} \|_2 \right) \\
+ w_2 \text{CE} \left( Y_{\text{stop}}, \hat{Y}_{\text{stop}}^{\text{bwd}} \right) \tag{17} \]

\[ + w_3 \left( \text{CE} \left( Y_{\text{outP}}, \hat{Y}_{\text{outP}}^{\text{bwd}} \right) \right). \]

Additionally, to force the two decoders to learn complementary information from each other, we train the two decoders to produce the same attention weights by including the following loss term,

\[ L_{\text{att}} = w_4 \| \alpha_{\text{fwd}} - \alpha_{\text{bwd}} \|_2, \tag{18} \]

where \( \alpha_{\text{fwd}} \) and \( \alpha_{\text{bwd}} \) are the attention weights of the forward and backward decoder, respectively.

The final loss term of the proposed system is,

\[ L_{\text{proposed}} = L_{\text{base}} + L_{\text{bwd}} + L_{\text{att}}. \tag{19} \]

The rationale behind the forward-and-backward decoding is that RNNs are generally more accurate at the initial decoding time steps, but performance decreases as the predicted sequence becomes longer because the prediction errors accumulate due to the autoregression. By including two decoders that model the input data in two different directions, and by constraining them to produce similar attention weights, we force the two decoders to incorporate information from both the past and future, thus improving their modeling power. Note that we only use both decoders during training. During inference time, we keep either the forward or backward decoder and discard the other. Therefore, the model size is exactly the same as the baseline model.

D. WaveGlow Vocoder

We use a WaveGlow vocoder [15] to convert the output of the speech synthesizer back into a speech waveform. WaveGlow is a flow-based [54] network capable of generating high-quality speech from Mel-spectrograms. It takes samples from a zero mean spherical Gaussian with variance \( \sigma \) with the same number of dimensions as the desired output and passes those samples through a series of layers that transform the simple distribution to one that has the desired distribution. In the case of training a vocoder, we use WaveGlow to model the distribution of audio samples conditioned on a Mel-spectrogram. During inference, random samples from the zero-mean spherical Gaussian are concatenated with the up-sampled (matching the speech sampling rate) Mel-spectrogram to predict the audio samples. WaveGlow can achieve real-time inference speed, whereas WaveNet takes a long time to synthesize an utterance due to its auto-regressive nature. For more details about the WaveGlow vocoder, we refer readers to the original study by Prenger et al. [15], which also showed that WaveGlow generates speech with quality comparable to WaveNet.

IV. EXPERIMENTAL SETUP

For the FAC task (training the speech synthesizers, WaveGlow neural vocoders, and pronunciation-correction models), we used one native speaker (BDL; American accent)\(^8\) from CMU-ARCTIC corpus [55] and two non-native speakers (YKWK, Korean; TXHC, Chinese) from the L2-ARCTIC corpus\(^9\) [56]. We split the data from all speakers into non-overlapping training (1032 utterances), validation (50 utterances), and testing (50 utterances) sets. Recordings from BDL were sampled at 16 kHz. Recordings in the L2-ARCTIC corpus were resampled from 44.1 kHz to 16 kHz to match BDL’s sampling rate and were pre-processed with Audacity [57] to remove any ambient noise.

\(^8\)We chose to use BDL as the native speaker since our AM has reasonable recognition accuracy on his speech (cf. Table I). If the AM were to perform poorly on the native speaker, then the L1-GS utterances would include more mispronunciations and therefore degrade the overall accent conversion performance.

\(^9\)[Online]. Available: https://psi.engr.tamu.edu/l2-arctic-corpus
background noise. In all FAC tasks, we extracted 80-dim Mel-spectrogram with a 10 ms shift and 64 ms window size. All neural network models were implemented in PyTorch [58] and trained with an NVIDIA Tesla P100 GPU. In all experiments, we trained speaker-dependent WaveGlow neural vocoders for L2 speakers using the official implementation provided by Prenger et al. [15].

V. EXPERIMENTS AND RESULTS

We conducted two experiments to evaluate the proposed FAC system on a thorough set of objective measures (e.g., word error rates, Mel Cepstral distortion) and subjective measures (degree of foreign accent, audio quality, and voice similarity). In experiment 1, we evaluated the reference-based golden speaker (L1-GS) generated by the L2 speech synthesizer (Section III-B). Then, in experiment 2, we evaluated the reference-free golden speaker (L2-GS) produced by the pronunciation-correction model (Section III-C).

A. Experiment 1: Evaluating the Reference-Based Golden Speaker (L1-GS)

We constructed the following three systems and compared their performance in generating L1-GS utterances. The objectives of this experiment were to determine the optimal speech embedding, and more importantly, to establish that L1-GS utterances captured the native accent and the L2 speaker identity, which is critical since they would be used as targets for the reference-free FAC task. Details of the model configurations and training are summarized in Appendix A.

- **Senone-PPG**: use the senone-PPG as the input (6024 dimensions).
- **Mono-PPG**: use the monophone PPG as the input (346 dimensions).
- **BNF**: use the bottleneck feature vector as the input (256 dimensions).

To generate the L1-GS utterances for testing, we extracted the three speech embeddings from speaker BDL’s test set and drove the systems with their respective input. The output Mel-spectrograms were then converted to speech through the WaveGlow vocoders.

1) **Objective Evaluation**: In a first experiment, we computed the word error rate (WER) of L1-GS utterances synthesized using each of the three speaker embeddings. In our case, the speech recognizer consisted of the TDNN-F acoustic model combined with an unpruned 3-gram language model trained on the Librispeech transcripts. As a reference, we also computed WERs on test utterances from the L1 speaker (BDL) and the two L2 speakers (YKWK, TXHC). Results are summarized in Table I. L1-GS utterances from the three systems achieve lower WERs than the corresponding utterances from the L2 speakers. Since the acoustic model had been trained on American English speech, a reduction in lower WERs can be interpreted as a reduction in the foreign-accentedness. The BNF system performs markedly better than the other two systems, achieving WERs that are close to those on L1 utterances. The Senone-PPG system performed the worst, despite the fact that it contains the most fine-grained triphone-level phonetic information. We offer an explanation of this result in the discussion.

2) **Subjective Evaluation**: To further evaluate the three L1-GS systems, we conducted formal listening tests to rate three perceptual attributes of the synthesized speech: accentuatedness, acoustic quality, and voice similarity. All listening tests were conducted through the Amazon Mechanical Turk platform. Instructions were given in each test to help the participants focus on the target speech attribute. All tests included five calibration samples to detect cheating behaviors, as suggested by Buchholz and Latorre [59]; responses from participants who were deemed to have cheated were excluded. Ratings for the calibration samples were excluded, too. All participants received monetary compensation. All samples were randomly selected from the test set, and the presentation order of samples in every listening test was randomized and counter-balanced. All participants resided in the United States at the time of the recruitment and passed a qualification test where they identified several regional dialects in the United States. All participants were self-reported native English speakers. AQAll listening tests in this study have been approved by the Institutional Review Board of Texas A&M University.

**Accentenedness test.** Listeners were asked to rate the foreign accent of an utterance on a nine-point Likert-scale (1: no foreign accent; 9: heavily accented), which is used in the pronunciation training community [60]. Listeners were told that the native accent in this task was General American. Participants (N = 20) rated 20 randomly selected utterances per system per L2 speaker. The utterances shared the same linguistic content in all conditions to ensure a fair comparison. As a reference, listeners also rated the same set of sentences for the L1 and L2 speakers. The results are summarized in the first row of Table II. L1-GS utterances from the three systems were rated significantly (p < 0.001) more native-like than the original L2 speech, though not as much as the original L1 speech. Among the three systems, the BNF system significantly outperformed...
Mono-PPG, while Mono-PPG was rated significantly more native-like than Senone-PPG, all with $p < 0.001$.

Acoustic quality. Listeners were asked to rate the acoustic quality of an utterance using a standard five-point (1: poor; 2: bad; 3: fair; 4: good; 5: excellent) Mean Opinion Score (MOS) [61]. Participants ($N = 20$) listened to 20 randomly-selected sentences per L2 speaker per system. As in the accent-ness test, listeners also rated the original utterances from the L1 and L2 speakers. The results are summarized in the second row of Table II. As expected, the original native speech received the highest MOS. Among the three golden speaker voices, BNF achieved the highest MOS compared with the other two systems ($p < 0.001$). The Mono-PPG system obtained better acoustic quality than the Senone-PPG system ($p = 0.045$). Interestingly, L1-GS utterances from the BNF system received higher MOS than the original L2 speech (3.78 vs. 3.70, $p = 0.02$), a surprising result for which we offer a possible explanation in Section VI.

Voice similarity test. Listeners were presented with a pair of speech samples—an L1-GS synthesis, and the original utterance from the corresponding L2 speaker. In the test, listeners first had to decide if the two samples were from the same speaker, and then rate their confidence level on a seven-point scale (1: not confident at all; 3: somewhat confident; 5: quite a bit confident; 7: extremely confident) [1], [27]. To minimize the influence of accent, the two utterances had different linguistic contents and were played in reverse, following [1]. For each system, participants ($N = 20$) rated 10 utterance pairs per speaker (20 utterance pairs for each system). Results are summarized in Table III. Across the three systems, more than 70% of the listeners were “quite a bit” confident (4.82-4.93 out of 7) that the L1-GS utterance and the original L2 utterance had the same voice identity. Significance tests showed that there was no statistically significant difference between the preference percentages for the three systems.

These results show that the BNF system outperforms the other two systems significantly in both objective and subjective measures. Therefore, for the remainder of this manuscript, we focus our evaluation on the BNF system, i.e., target L1-GS utterances for the reference-free ( pronunciation-correction) system are those from the BNF system.

B. Experiment 2: Evaluating the Reference-Free Golden Speaker (L2-GS)

In the second experiment, we directly converted L2 test utterances with the proposed pronunciation-correction model and compared it against the baseline systems. Detailed model architecture configurations and training setups are included in Appendix B.

- **Baseline 1**: the system of Zhang et al. [12], a state-of-the-art VC system capable of modifying segmental and prosodic attributes between different speakers. The loss function of this system was eq. (16), i.e., $L_{base}$.

- **Baseline 2**: the system of Liu et al. [41], the only other reference-free accent conversion system that we are aware of (cf. Section II-C). The audio samples were generated by passing the test set utterances through the Liu system, as prescribed in [41], which was pre-trained on 105 VCTK [62] speakers. The test samples were provided as a courtesy by Liu et al. and we only performed two post-processing steps to ensure a fair comparison. First, we resampled the test samples provided by Liu et al. from 22.05 kHz to 16 kHz to match the sampling rate of the other systems. Second, we manually trimmed the trailing white noises in some of the test samples. The accent conversion model was pre-trained on VCTK not L2-ARCTIC, which made its stop-token prediction not stable, and some of the synthesized utterances have a few seconds of white noise after the end of speech.

- **Proposed ( without att loss)**: the proposed system without the attention loss term described in eq. (18). We included this variation of the proposed system to study the contribution of adding the backward decoder alone. The loss function of this system was $L_{base} + L_{bwd}$.

- **Proposed**: the proposed system with the full forward-and-backward decoding technique, which included both the backward decoder and the attention loss term. The loss function of this system was eq. (19), i.e., $L_{base} + L_{bwd} + L_{att}$.

For both variations of the proposed system, we performed accent conversion using the backward decoder during testing since it produced significantly better-quality speech compared to the forward decoder on the validation set. Please refer to Appendix C for a qualitative comparison between the two decoders.

1) Objective Evaluations: For objective evaluations, we computed three measures, as suggested by [12], plus WER as a fourth:

- **MCD**: the Mel-Cepstral Distortion [28] between the L2-GS (actual output) and L1-GS speech (desired output). It was computed on time-aligned (Dynamic Time Warping) Mel-cepstra between the L2-GS and the L1-GS audio. Lower MCD correlates with better spectral predictions. We used SPTK [63] and the WORLD vocoder [64] to extract the Mel-cepstra with a shift size of 10 ms.

- **$F_0$ RMSE**: the $F_0$ RMSE between the L2-GS and L1-GS speech on voiced frames. Lower $F_0$ RMSE represents better pitch conversion performance. The $F_0$ and voicing features were extracted by the WORLD vocoder with the Harvest pitch tracker [65].

- **DDUR**: the absolute difference in duration between the L2-GS and L1-GS speech. Lower DDUR implies better duration conversion performance.

- **WER**: the word error rate for the L2-GS speech. Ideally, the L2-GS speech should have a lower WER than the original
TABLE IV

<table>
<thead>
<tr>
<th>L2 speaker</th>
<th>System</th>
<th>WER (%)</th>
<th>MCD (dB)</th>
<th>F0 RMSE (Hz)</th>
<th>DDUR (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>YKWK</td>
<td>Original</td>
<td>45.82</td>
<td>8.07</td>
<td>23.38</td>
<td>1.15</td>
</tr>
<tr>
<td></td>
<td>Baseline 1</td>
<td>41.31</td>
<td>6.26</td>
<td>18.43</td>
<td>0.18</td>
</tr>
<tr>
<td></td>
<td>Baseline 2</td>
<td>82.81</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>Proposed (w/o att loss)</td>
<td>36.12</td>
<td>6.16</td>
<td>19.41</td>
<td>0.14</td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>34.54</td>
<td>6.10</td>
<td>20.78</td>
<td>0.15</td>
</tr>
<tr>
<td>L1-GS</td>
<td>9.50</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>TXHC</td>
<td>Original</td>
<td>44.57</td>
<td>8.00</td>
<td>25.73</td>
<td>1.29</td>
</tr>
<tr>
<td></td>
<td>Baseline 1</td>
<td>43.67</td>
<td>6.32</td>
<td>19.40</td>
<td>0.17</td>
</tr>
<tr>
<td></td>
<td>Baseline 2</td>
<td>84.39</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>Proposed (w/o att loss)</td>
<td>40.05</td>
<td>6.26</td>
<td>22.33</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>37.33</td>
<td>6.29</td>
<td>21.37</td>
<td>0.15</td>
</tr>
<tr>
<td>L1-GS</td>
<td>7.47</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Original</td>
<td>45.20</td>
<td>8.04</td>
<td>24.56</td>
<td>1.22</td>
</tr>
<tr>
<td></td>
<td>Baseline 1</td>
<td>42.49</td>
<td>6.29</td>
<td>18.92</td>
<td>0.18</td>
</tr>
<tr>
<td></td>
<td>Baseline 2</td>
<td>83.60</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>Proposed (w/o att loss)</td>
<td>38.09</td>
<td>6.21</td>
<td>20.87</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>35.94</td>
<td>6.20</td>
<td>21.08</td>
<td>0.15</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>4.49</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

non-native speech, implying that the conversion reduced the foreign accent.

Results are summarized in Table IV. For all measures, we also computed the scores between the original L2 speech and the L1-GS speech as a reference. In addition, we included the WER of the L1-GS speech as an upper-bound. By definition, the other three measures on the L1-GS speech are all zero. For Baseline 2, we only computed the WER since the system was not trained to predict L1-GS, which makes computing the other objective scores ill-defined.

The two variations of the proposed method obtained better WER, MCD, and DDUR scores, while the Baseline 1 method performed slightly better on the $F_0$ RMSE. More importantly, Baseline 1 and the two variations of the proposed method were able to reduce the WER of the input L2 utterance. The Proposed method (with attention loss) reduced WERs by 20.5% (relative) on average, which was significantly higher than the WER reduction of the Baseline 1 system (6.0% relative). Baseline 2 performed poorly on the WER metric. Among the two variations of the proposed method, the one that included both the backward decoder and attention loss performed equally-well or better on the WER, MCD, and DDUR scores.

2) Subjective Evaluations: Following the same protocol described in Section V-A2, we asked participants to rate the accentness, acoustic quality, and voice similarity of synthesized L2-GS utterances. We used the samples from the Proposed system (with the attention loss during training) based on the objective evaluations in the previous section.

Accentness test. Participants (N = 20) rated 20 random samples per speaker per system, as well as the corresponding original audio. Results are compiled in the first row of Table V. All systems obtained significantly more native-like ratings than the original L2 utterances ($p < 0.001$). More specifically, the Baseline 1 system reduced the accentness rating by 15.5% (relative) and the Baseline 2 system reduced the accentness rating by 8.2% (relative), while the Proposed system achieved a 19.0% relative reduction, a difference that was statistically significant (Proposed and Baseline 1, $p = 0.04$; Proposed and Baseline 2, $p < 0.001$). As expected, the original L1 speech was rated less accented than all other systems.

MOS test. Participants (N = 20) rated 20 audio samples per speaker per system. We used the same MOS test as in experiment 1 to measure the acoustic quality of the synthesis. Results are shown in the second row of Table V. The Proposed system achieved significantly better audio quality than the baselines (9.15% relative improvement compared with Baseline 1; 12.59% relative improvement compared with Baseline 2; $p < 0.001$ in both cases).

Voice similarity test. Participants (N = 20) rated 10 utterance pairs per speaker per system (i.e., 20 utterance pairs for each system). This last experiment verified that the accent conversion retained the voice identity of the L2 speakers. The results are shown in Table VI. For Baseline 1 and the Proposed system, the majority of the participants thought the synthesis and the reference speech were from the same speaker, and they were “quite a bit confident” (5.00-5.12 out of 7) about their ratings. Although the Proposed system obtained higher ratings than the Baseline 1 system in terms of voice identity, the difference between the preference percentages was not statistically significant ($p = 0.12$), which was expected. The reason is that the input and output speech had different accents, but very similar voice identity. Therefore, both systems were not trained to modify the voice identity of the input audio. As a result, both the Baseline 1 system and the Proposed system were able to keep the voice identity unaltered during the conversion process. The Baseline 2 system, on the other hand, performed significantly worse than Baseline 1 and the Proposed system in terms of voice similarity; on average, 47.5% of the participants thought that the synthesis and the reference speech were from the same speaker, which is lower than chance level, indicating that the syntheses produced by Baseline 2 did not capture the voice identity of the L2 speakers.
well. This result echoes with the findings of Liu et al. [41], where they also identified voice identify issues of the Baseline 2 system.

Aside from the objective and subjective scores, we provide an example of the attention weights produced by Baseline 1 and the Proposed system on a test utterance in Figure 6. Qualitatively, we can observe that the attention weights of the Baseline 1 system contained an abnormal jump towards the end of the synthesis, while the Proposed system produced smooth alignments at the same time steps. Additionally, the Proposed method appears to have used a broader window to compute the attention context compared with Baseline 1, as reflected by the width of the attention alignment path. Therefore, the Proposed system utilized more contextual information during the decoding process.

VI. DISCUSSION

A. Experiment 1

In experiment 1, we tested three versions of the L1-GS system that used different speech embeddings at the input: senone-PPGs, monophone-PPGs, and bottleneck features (BNFs). Both objective and subjective tests suggested that the BNF system outperforms the other two, both in terms of audio quality and native accentedness. Further, we find that L1-GS utterances on the BNF system achieve similar WERs as the original utterances from the L1 speaker, a remarkable result that further supports the effectiveness of the system in reducing foreign accents. The majority of the human raters (73.75%) had high confidence that the BNF L1-GS shared the same voice identity as the target L2 speaker, suggesting that the accent conversion was also able to preserve the desired (i.e., the L2 speaker’s) voice identity. A surprising result from the listening tests is that BNF L1-GS utterances were rated to have higher audio quality than the original natural speech from the L2 speaker. Although this result speaks of the high acoustic quality that the BNF L1-GS system is able to achieve, it is likely that native listeners associated acoustic quality with intelligibility, rating the original foreign-accented speech to be of lower acoustic quality because of that; see Felps et al. [1].

Two probable factors explain why BNFs outperformed the other two speech embeddings. First, during the training process, we observed that BNF L1-GS utterances were rated to have higher audio quality than the original natural speech from the L2 speaker. Although this result speaks of the high acoustic quality that the BNF L1-GS system is able to achieve, it is likely that native listeners associated acoustic quality with intelligibility, rating the original foreign-accented speech to be of lower acoustic quality because of that; see Felps et al. [1].

B. Experiment 2

In experiment 2, we achieved reference-free FAC by constructing a pronunciation-correction model that converted L2 utterances directly to match the L1-GS. Our results are encouraging; both the baseline model of Zhang et al. [12] (Baseline 1)
and our proposed system were able to reduce the foreign accentedness of the input speech significantly, while retaining the voice identity of the L2 speaker. More importantly, the proposed system outperformed the Baseline 1 system significantly in terms of MOS and accentedness ratings. A possible explanation for this result is that the proposed method computes the alignment between each pair of input and output sequences from two directions at training time. Thus, by forcing the forward and the backward decoders to produce similar alignment weights, we force the decoders to incorporate information from both the past and future when generating the alignment. During inference time, only one decoder is needed to perform the reference-free accent conversion; therefore, the proposed system consumes exactly the same amount of inference resources as the baseline system. In summary, the better accentedness and audio quality ratings obtained by the proposed system can largely be attributed to the better alignments provided by the forward-and-backward decoding training technique, as illustrated in Figure 6. The proposed system also outperformed a state-of-the-art reference-free FAC system by Liu et al. [41] (Baseline 2) in all objective and subjective evaluation metrics. The comparison of the proposed method and Baseline 2 shows that there is still a large performance gap between a speaker-specific reference-free FAC system (the proposed method) and a many-to-many reference-free FAC system (Baseline 2), which encourages future work in both areas.

The L2-GS generated by the reference-free FAC was rated as significantly less accented than the L2 speaker, though it still had a noticeable foreign accent compared with the original L1 speech. This suggests that the pronunciation-correction model did not fully eliminate the foreign accent in heavily mispronounced or disfluent speech segments, and therefore some foreign-accent cues from the input were carried over to the output speech. One likely explanation for this result is that the proposed reference-free FAC model can only correct error patterns that have occurred in the training data. Due to the high variability of L2 pronunciations, the amount of training data available for each L2 speaker (~one hour of speech) was not sufficient to cover a portion of the error patterns manifested in the test data, and therefore those errors were not corrected and resulted in the residual foreign accents in the L2-GS utterances. Finally, the MOS ratings of the pronunciation-correction models were lower than those of the BNF L1-GS, which was expected since the output of the pronunciation-correction model is a re-synthesis of the L1-GS utterances.

VII. CONCLUSION AND FUTURE WORK

In this work, we propose a new reference-free FAC system\textsuperscript{12} that transforms input L2 utterances to reduce their foreign accentedness. This is in contrast to the majority of the existing FAC systems, which require native reference utterances at inference time. Training the system requires two steps. In a first step, we train a FAC model to transform utterances from a reference L1 speaker, so they have the voice identity of the L2 learner. We refer to these transformed utterances as L1-GS utterances. In a second step, we train a pronunciation-correction model that can transform utterances from the L2 learner to match the L1-GS utterances obtained in the first step. Our evaluations indicate that the reference-free FAC system can significantly reduce the foreign accentedness in L2 speech while retaining the voice identity.

One possible future direction of this work is to use transfer learning \cite{66} to reduce the amount of training data needed for the golden-speaker generation process. This would require first training a multi-speaker speech synthesizer with speech embeddings and speaker embeddings (e.g., i-vectors) as the input, then performing inference using speech embeddings from the reference L1 speaker and the speaker embeddings from the L2 speaker. The benefit of this strategy is that training a multi-speaker speech synthesizer generally only requires a small number of recordings from a particular speaker (e.g., the L2 speaker).

Another future research direction is to improve the quality of the pronunciation-correction model. A direct extension of the current system that might improve the audio quality is to jointly optimize the pronunciation-correction model and the neural vocoder. The current setup of the system trains the WaveGlow model with “clean” original Mel-spectrograms, which leads to a mismatch between the output of the pronunciation-correction model (synthetic Mel-spectrogram) and the expected input of the neural vocoder. Another possibility for quality improvement is to directly convert between foreign-accented and native speech embeddings to correct the mispronunciations. This seems feasible since the speech embeddings (e.g., BNF) contain rich classifiable phonetic information, which is decoupled from other speaker-specific cues that might interfere with the correction process. The benefits of this approach are two-fold. First, it would eliminate the need to generate the L1-GS, since we can directly use the speech embeddings from L1 teachers as training targets. Second, by combining data from speakers that share the same foreign accent, this approach would enable us to train specific pronunciation-correction models for each first language (e.g., for Chinese L2 learners of English) that can cover more mispronunciation variations compared with speaker-dependent models, as we have done in this current work, thus improving the accentedness ratings of the syntheses. AQFinally, we intend to study other simpler attention regularization techniques \cite{67} as alternatives to the forward-and-backward decoding technique used in this work. A simpler attention regularization technique would help the pronunciation-correction model lower its training cost.

APPENDIX A

MODEL DETAILS OF THE SPEECH SYNTHESIZERS

Table VII summarizes the neural network architectures of the three speech synthesizers. It is worth noting that the input PreNet produced a 512-dim summarization from the Senone-PPG, which is higher than the dimensionality of the Mono-PPG and BNF. We did experiment on a lower dimensionality (256) in the input PreNet, which lead to significant artifacts and

\textsuperscript{12}Project webpage: https://guanlongzhao.github.io/demo/reference-free-ac
mispronunciations. Therefore, we used the current setting for the Senone-PPG system in order to generate intelligible speech syntheses to compare with the other two systems.

The models were trained using the Adam optimizer [68] with a constant learning rate of $1 \times 10^{-4}$ until convergence, which was monitored by the validation loss. We applied a $1 \times 10^{-6}$ weight decay [69] and a gradient clipping [70] of 1.0 during training. The batch size was set to 8 and the weight terms $w_1$ and $w_2$ in eq. (13) were set to 1.0 and 0.005, based on preliminary experiments [27].

**APPENDIX B**

**MODEL DETAILS OF THE PRONUNCIATION-CORRECTION MODELS**

Table VIII summarizes the model details of the Baseline 1 pronunciation-correction model. On top of the Baseline 1 model, the Proposed model adds a backward decoder that has the same structure (attention modules, decoder LSTM, and decoder PreNet) as the Baseline 1 model’s decoder. The phoneme prediction ground-truth labels were per-frame phoneme labels (with word positions) that were produced by force-aligning the audio to its orthographic transcriptions. We note that the phoneme predictions were only required in training, not testing. For both models, the training was performed with the Adam optimizer with a weight decay of $1 \times 10^{-6}$ and a gradient clip of 1.0. The initial learning rate was $1 \times 10^{-3}$ and was kept constant for the first 20 epochs, then exponentially decreased by a factor of 0.99 at each epoch for the next 280 epochs, and then kept constant at the terminal learning rate. The batch size was 16. The loss term weights $w_1$, $w_2$, $w_3$, and $w_4$ in equations (16)-(19) were empirically set to 1.0, 0.05, 0.5, and 100.0.

**APPENDIX C**

**QUANTITATIVE COMPARISON BETWEEN THE FORWARD AND BACKWARD DECODER OF THE PROPOSED SYSTEM**

As a qualitative comparison between the forward and backward decoder in the proposed system, we plot the attention weights generated by both decoders on a few utterances from the validation set. Good alignment of the attention weights generally indicates better performance. We can see in the figures that the backward decoder produces attention weights that have less discontinuity, which may explain why the backward decoder generates speech with better quality compared to the forward decoder.

**ACKNOWLEDGMENT**

We would like to thank Dennis R. da Cunha Silva, Adam Hair, and Pedro Moreno for reviewing early versions of the manuscript. We appreciate the help of Songxiang Liu (The Chinese University of Hong Kong) for providing test samples for Baseline 2. We are grateful for the feedback from the reviewers, which helped improve the quality of the manuscript.

**REFERENCES**


Guanlong Zhao received the B.S. degree in applied physics from the University of Science and Technology of China, Hefei, China, in 2015 and the Ph.D. degree in computer science from Texas A&M University, College Station, TX, USA, in 2020. He is currently a Software Engineer with Google, where he is working on speech recognition model training infrastructure, automation, and internationalization. His research interests include speech synthesis, acoustic modeling, voice conversion, and accent conversion.

Shaojin Ding received the B.S. degree in automation from Xi’an Jiaotong University, Xi’an, China, in 2015. He is currently working toward the Ph.D. degree in computer science with Texas A&M University, College Station, TX, USA. His research interests include speech synthesis, voice conversion, and speaker recognition.

Ricardo Gutierrez-Osuna (Senior Member, IEEE) received the B.S. degree in electrical engineering from the Polytechnic University of Madrid, Madrid, Spain, in 1992, and the M.S. and Ph.D. degrees in computer engineering from North Carolina State University, Raleigh, NC, USA, in 1995 and 1998, respectively. He is currently a Professor with the Department of Computer Science and Engineering, Texas A&M University, College Station, TX, USA. His current research interests include voice and accent conversion, speech and face perception, wearable physiological sensors, and active sensing.