Towards End-to-End Prosody Transfer for Expressive Speech Synthesis with Tacotron

RJ Skerry-Ryan [1] Eric Battenberg [1] Ying Xiao [1] Yuxuan Wang [1] Daisy Stanton [1] Joel Shor [1] Ron J. Weiss [1]
Rob Clark [1] Rif A. Saurous [1]

Abstract

We present an extension to the Tacotron speech synthesis architecture that learns a latent embedding space of prosody, derived from a reference acoustic representation containing the desired prosody. We show that conditioning Tacotron on this learned embedding space results in synthesized audio that matches the prosody of the reference signal with fine time detail even when the reference and synthesis speakers are different. Additionally, we show that a reference prosody embedding can be used to synthesize text that is different from that of the reference utterance. We define several quantitative and subjective metrics for evaluating prosody transfer, and report results with accompanying audio samples from single-speaker and 44-speaker Tacotron models on a prosody transfer task.

1. Introduction

In order to produce realistic speech, a text-to-speech (TTS) system must implicitly or explicitly impute many factors that are not provided by simple text input. Such factors include the intonation, stress, rhythm and style of the speech, and are collectively referred to as *prosody*.

Speech synthesis via text-to-speech is a challenging underdetermined problem, since the meaning expressed by an utterance is inherently underspecified by the text. For example, the simple statement "The cat sat on the mat." can be spoken many different ways. If the statement is the answer to the question "Where did the cat sit?" the speaker might stress the word "mat" to indicate that it is the answer to the question. To express uncertainty in their knowledge, the speaker may decide to intone the response with a rising pitch. The question, "Would you like an apple or an orange?" can also be spoken in multiple ways, indicating information about the set of objects that exist. If there are only two possible options, the intonation of the final option ("orange") will have a declining pitch. If there are a variety of options of which apple and orange are just two examples, both options are typically intoned with a rising pitch. The intonation of these sentences carries meaning about the environment or context of the question which is unspecified by the text, and in general, there are any number of such nuances present in speech that convey information beyond the textual content.

In order to avoid the challenging problem of schematizing and labeling prosody, we seek methods of modeling prosody that do not require explicit annotations, and present an architecture for learning a latent prosody representation by extracting it from the ground truth speech audio. Accordingly, we use a "subtractive" definition of prosody:

Definition. *Prosody is the variation in speech signals that remains after accounting for variation due to phonetics, speaker identity, and channel effects (i.e. the recording environment).*

This view of prosody is compatible with interpretations of prosody from previous works (Wagner & Watson, 2010; Ladd, 2008).

One natural problem that arises from this formulation is *sampling* – that is, the challenge of generating diverse and interesting prosody and output speech even for identical phonetics, speaker identities, and channel effects. In this paper, we tackle the more basic problem of *constructing* a space that represents prosody. We propose one possible construction of a prosody latent space, and show that we capture meaningful variation in speech by demonstrating transfer in this space (i.e., using a latent representation to make one utterance sound like another): this roughly corresponds to a "say it like this" task.

The recently proposed Tacotron speech synthesis system (Wang et al., 2017a) computes its output directly from graphemes or phonemes, and its prosody model is implicit, learned from the statistics of the training data alone. It learns, for example, that an English sentence ending in a question mark likely has a rising pitch if the question has a

[1] Google, Inc.. Correspondence to: RJ Skerry-Ryan <rjryan@google.com>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

yes-or-no answer. In this work[1], we augment Tacotron with explicit prosody controls. We accomplish this by learning an encoder architecture that computes a low-dimensional embedding from a speech signal, where the embedding provides information not provided by the text and speaker identity. Through careful experiments, we demonstrate that this prosody embedding can be used to reproduce the desired prosody using Tacotron.

The immediate implication of this acoustic encoder architecture and prosody latent space is that we can control the behavior of a TTS system using a different voice than the one used in training. The resulting embedding is fixed-length and often smaller than the transcript, so it can be easily stored alongside the text for use in a production system. The longer-term implications are that we can build models that predict prosody embeddings from non-acoustic context, such as prosody labels or conversation state.

Our main contribution is an encoder architecture that extracts a fixed-length learned representation of prosody from acoustic input; we demonstrate that this encoder allows us to *transfer* prosody between utterances containing similar text in an almost speaker-independent fashion. To evaluate performance in this prosody transfer task, we propose a number of quantitative and qualitative metrics. Additionally, we strongly encourage the reader to listen to the audio samples on our demo page.

2. Related Work

The modeling of prosody and speaking style has been investigated since the era of HMM-based TTS research. For example, (Eyben et al., 2012) proposes a system that first clusters the training set, and then performs HMM-based cluster-adaptive training. (Nose et al., 2007) proposes estimating a transformation matrix for a set of predefined style vectors.

Numerous works have explored annotation schemes for diagramming and automatic labeling of prosody: ToBi (Silverman et al., 1992), AuToBI (Rosenberg, 2010), Tilt (Taylor, 1998), INTSINT (Hirst, 2001), and SLAM (Obin et al., 2014) all describe methods for the annotation and automatic extraction of labels or annotations that correlate with prosodic phenomena. The challenges of annotation often require domain experts, however, and inter-rater annotations can differ substantially (Wightman, 2002).

Few works propose the use of acoustic reference signals to control the prosody of a text-to-speech model. (Tesser et al., 2013) proposes the use of "signal driven" features to predict symbolic prosody representations, using AuToBI labels to improve HMM-based synthesis. (Coile et al., 1994) propose "prosody transplantation" via a system called PROTRAN for recording a low-bit-rate "enriched phonetic transcription" that can be used in conjunction with desired text to reproduce the prosody of an original recording. Note that the same product needs described in (Coile et al., 1994) motivate the development of this paper.

Prosody transfer is related to the task of voice conversion (also called style transfer in the audio context). To perform voice conversion, a model must synthesize an utterance, given only the acoustic signal of that utterance in a different speaker's voice (Wu et al., 2013; Nakashika et al., 2016; Kinnunen et al., 2017; van den Oord et al., 2017; Chorowski et al., 2017). An approach similar to ours can be found in (Wang et al., 2018), where a more complicated autoencoder is used to learn some elements of style in an unsupervised fashion.

3. Model Architecture

Our model is based on Tacotron (Wang et al., 2017a), a recently proposed state-of-the-art end-to-end speech synthesis model that predicts mel spectrograms directly from grapheme or phoneme sequences. The predicted mel spectrograms can either be synthesized directly to the time-domain via a WaveNet vocoder (Shen et al., 2017), or by first learning a linear spectrogram prediction network, and then applying Griffin-Lim spectrogram inversion (Griffin & Lim, 1984).

In this work, we use the original encoder and decoder architecture from (Wang et al., 2017a), not the simplified architecture proposed by (Shen et al., 2017). Additionally, we exclusively use phoneme inputs produced by a text normalization front-end and lexicon, as we are specifically interested in addressing prosody, not the model's ability to learn pronunciation from graphemes. Finally, instead of the Bahdanau content-based attention used in (Wang et al., 2017a), we use the GMM attention of (Graves, 2013) which we find improves generalization to long utterances.

The audio samples included on our demo page were produced with a WaveNet vocoder (Shen et al., 2017); however, the original linear-spectrogram prediction network followed by Griffin-Lim spectrogram inversion from (Wang et al., 2017a) works equally well for prosody transfer. In practice, we find the choice of neural vocoder only impacts audio fidelity and has no impact on the system's resulting prosody.

3.1. Multi-speaker Tacotron

Tacotron as proposed in (Wang et al., 2017a) does not include explicit modeling of speaker identity; however, due to the flexibility of all-neural sequence-to-sequence models, learning multi-speaker models via speaker identity condi-

[1] Sound demos are available at `https://google.github.io/tacotron/publications/end_to_end_prosody_transfer`.

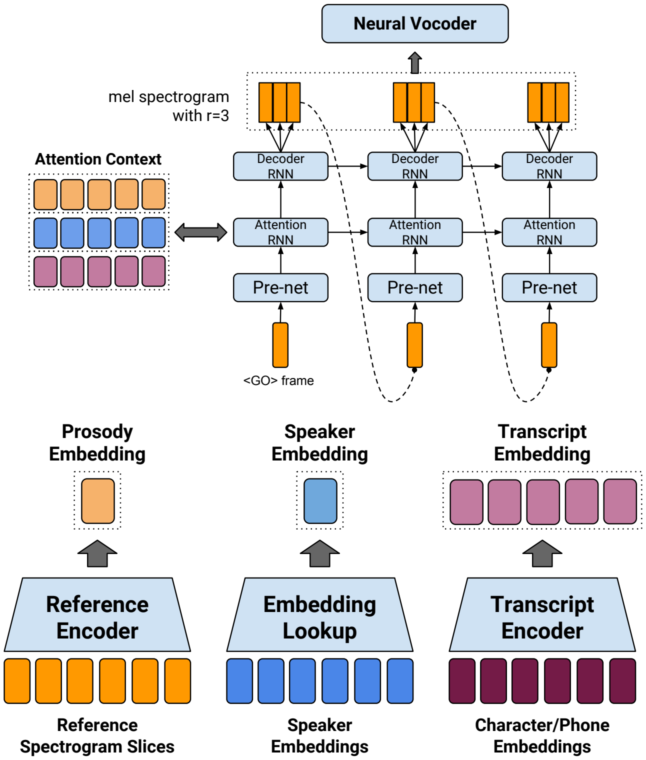

Figure 1. The full Tacotron architecture for prosody control. The autoregressive decoder is conditioned on the result of the reference encoder, transcript encoder, and speaker embedding via an attention module.

tioning is straightforward. We follow a scheme similar to (Arık et al., 2017) to model multiple speakers.

The Tacotron architecture conditions an auto-regressive decoder on an $L_T \times d_T$-dimensional representation of the phoneme or grapheme sequence produced by a transcript encoder architecture, where L_T is the length of the encoded transcript representation (typically equal to the length of the input transcript) and d_T is the embedding dimension produced by the transcript encoder. For each speaker in the dataset, an $\mathbb{R}^{d_S}$ embedding vector is initialized with Glorot (Glorot & Bengio, 2010) initialization. For each example, the d_S-dimensional speaker embedding corresponding to the true speaker of the example is broadcast-concatenated to the $L_T \times d_T$-dimensional transcript encoder representation to form a $(d_T + d_S)$-dimensional sequence of encoder embeddings that the decoder will attend to. No additional changes or loss metrics are necessary. For single-speaker datasets we do not use a speaker embedding.

3.2. Reference Encoder

We extend the Tacotron architecture by adding a "reference encoder" module that takes a length-L_R and d_R-dimensional reference signal as input, and computes a d_P-dimensional embedding from it. Instantiations of this fixed-dimensional embedding define a "prosody space" – our goal is that sampling from this space will yield diverse and plausible output speech, and that we can manipulate elements of this space to control the output meaningfully.

As with the speaker embedding, this prosody embedding is combined with the $L_T \times d_T$ text encoder representation via a broadcast-concatenation. In combination with the speaker embeddings described in Section 3.1, the encoder embeddings form a $L_T \times (d_T + d_S + d_P)$ embedding matrix, where the speaker and prosody embeddings are fixed across all timesteps. Figure 1 illustrates this structure.

During training, the reference acoustic signal is simply the target audio sequence being modeled. No explicit supervision signal is used to train the reference encoder; it is learned using Tacotron's reconstruction error as its only loss. In training, one can think of the combined system as an RNN encoder-decoder (Cho et al., 2014a) with phonetic and speaker information as conditioning input. For a sufficiently high-capacity embedding, this representation could simply learn to copy the input to the output during training. Therefore, as with an autoencoder, care must be taken

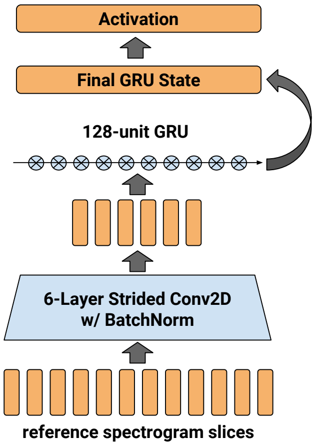
Activation
Final GRU State
128-unit GRU
6-Layer Strided Conv2D w/ BatchNorm
reference spectrogram slices

Figure 2. The prosody reference encoder module. A 6-layer stack of 2D convolutions with batch normalization, followed by "recurrent pooling" to summarize the variable length sequence, followed by an optional fully connected layer and activation.

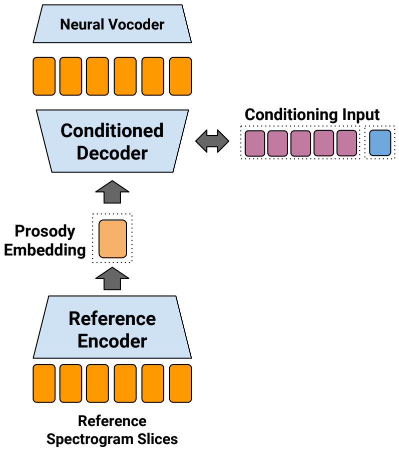

Figure 3. An interpretation of the Tacotron architecture for prosody control from Figure 1 as an RNN encoder-decoder with speaker and phonetic conditioning input.

to choose an architecture that sufficiently bottlenecks the prosody embedding such that it is forced to learn a compact representation.

During inference, we can use the prosody reference encoder to encode *any* utterance: we are not constrained to match either the text input or the speaker identity. In particular, this enables the possibility of prosody transfer – using an utterance by a different speaker, or different text to control the output. We study prosody transfer in detail in Section 4.

For the reference encoder architecture (Figure 2), we use a simple 6-layer convolutional network. Each layer is composed of 3×3 filters with 2×2 stride, using "same" padding and ReLU activations. Batch normalization (Ioffe & Szegedy, 2015) is applied to every layer. The number of filters in each layer doubles at half the rate of downsampling: 32, 32, 64, 64, 128, 128.

The $L_R \times d_R$ reference signal is downsampled by this architecture 64 times in both dimensions. The $\lceil d_R/64 \rceil$ feature dimensions and 128 channels of the final convolution layer are unrolled as the inner dimension of the resulting $\lceil L_R/64 \rceil \times 128 \lceil d_R/64 \rceil$ matrix. To compress the $\lceil L_R/64 \rceil$-length sequence produced by the CNN layers down to a single fixed-length vector, we use a recurrent neural network with a single 128-width Gated Recurrent Unit (GRU) (Cho et al., 2014b) layer. We take the final 128-dimensional output of the GRU as the pooled summarization of the sequence.

To compute the final d_P-dimensional embedding from the 128-dimensional output of the GRU, we apply a fully-connected layer to project the output to the desired dimensionality, followed by an activation function (e.g. softmax, tanh). The choice of activation function can constrain the information contained in the embedding and make learning easier by controlling its magnitude. After some exploration, we found that a d_P of 128 and a tanh activation perform well in practice.

3.3. Reference signal feature representation

The choice of $L_R \times d_R$ feature representation used as the input to the reference encoder architecture naturally impacts the aspects of prosody we can expect to learn. For example, a pitch track representation will not allow us to model prominence in some languages since it does not contain energy information. Similarly an MFCC representation may be somewhat pitch-invariant (depending on the number of coefficients retained), preventing us from modeling intonation. In this work, we decided to use the same perceptually-relevant summarization of the spectrum that (Wang et al., 2017a) does: the mel-warped spectrum (Stevens et al., 1937). As in (Wang et al., 2017a), we use 80 mel bands from 80 to 12000 Hz.

This choice of representation enables an interpretation of the resulting architecture as an RNN encoder-decoder (Cho et al., 2014a) conditioned on text and speaker identity. All it must model via its bottleneck representation is the unexplained variation in the signal, i.e. the prosody and recording environment. We illustrate this interpretation in Figure 3.

We also explored more compact representations, such as pitch track and intensity, but mel spectrograms produced the best results.

3.4. Variable Length Prosody Embeddings

The use of a fixed-length prosody embedding poses an obvious scaling bottleneck, preventing the extension of this approach to longer utterances. An alternate implementation of the reference encoder in Section 3.2 uses the output of the GRU at every time step rather than just the final output. As with the fixed-length encoder, each GRU output is passed through a fully connected layer to transform it to the desired dimensionality. This can be interpreted as a low-bitrate representation of prosody similar to the Enhanced Phonetic Transcriptions proposed in (Coile et al., 1994). To condition the Tacotron decoder on this sequence, we introduce a second attention head with an attention-aggregator module as proposed in (Wang et al., 2017b).

In our experiments, variable-length prosody embeddings are able to generalize to very long utterances; however, compared to fixed-length embeddings, variable-length embeddings are not as robust to text and speaker perturbations likely because they encode a stronger timing signal. Therefore, this paper focuses on fixed-length embeddings.

4. Experiments and Results

4.1. Datasets and training

We use the following datasets:

Single-speaker dataset: A single speaker high-quality English dataset of audiobook recordings by Catherine Byers (the speaker from the 2013 Blizzard Challenge). This dataset consists of 147 hours of recordings of 49 books, read in an animated and emotive storytelling style.

Multi-speaker dataset: A proprietary high-quality English speech dataset consisting of 296 hours across 44 speakers (5 with Australian accents, 6 with British accents, 1 with an Indian accent, 2 with Singaporean accents, and 30 with United States accents).

We train our models for at least 200k steps with a minibatch size of 256 using the Adam optimizer (Kingma & Ba, 2015). We start with a learning rate of 1×10^{-3} and decay it to 5×10^{-4}, 3×10^{-4}, 1×10^{-4}, and 5×10^{-5} at step 50k, 100k, 150k, and 200k respectively. For baselines, we train models without the reference encoder architecture (Section 3).

4.2. Evaluation metrics

There are no generally-accepted metrics for prosody transfer. To measure performance, we adapt a number of metrics from general audio processing, each of which reflects an acoustic correlate of prosody. For all comparisons of predicted signals to target signals, we extend the shorter signal to the length of the longer signal using a domain-appropriate padding (e.g. 0 for a time-domain waveform, -13.8 for a log magnitude spectrogram with a 1×10^{-6} stabilizing offset). All pitch and voicing metrics are computed using the output of the YIN (De Cheveigné & Kawahara, 2002) pitch tracking algorithm.

Mel Cepstral Distortion (MCD$_K$) (Kubichek, 1993):

$$\mathrm{MCD}_K = \frac{1}{T} \sum_{t=0}^{T-1} \sqrt{\sum_{k=1}^{K} \left(c_{t,k} - c'_{t,k}\right)^2}$$

Where $c_{t,k}$,$c'_{t,k}$ are the k-th mel frequency cepstral coefficient (MFCC) of the t-th frame from the reference and predicted audio. We sum the squared differences over the first K MFCCs, skipping $c_{t,0}$ (overall energy).

Gross Pitch Error (GPE) (Nakatani et al., 2008):

$$\mathrm{GPE} = \frac{\sum_t 1\left[|p_t - p'_t| > 0.2 p_t\right] 1[v_t] 1[v'_t]}{\sum_t 1[v_t] 1[v'_t]}$$

Where p_t,p'_t are the pitch signals from the reference and predicted audio, v_t,v'_t are the voicing decisions from the reference and predicted audio, and 1 is the indicator function. The GPE measures the percentage of voiced frames that deviate in pitch by more than 20% compared to the reference.

Voicing Decision Error (VDE) (Nakatani et al., 2008):

$$\mathrm{VDE} = \frac{\sum_{t=0}^{T-1} 1[v_t \neq v'_t]}{T}$$

Where v_t,v'_t are the voicing decisions for the reference and predicted audio, T is the total number of frames, and 1 is the indicator function.

F0 Frame Error (FFE) (Chu & Alwan, 2009):

$$\frac{\sum_{t=0}^{T-1} 1\left[|p_t - p'_t| > 0.2 p_t\right] 1[v_t] 1[v'_t] + 1[v_t \neq v'_t]}{T}$$

FFE measures the percentage of frames that either contain a 20% pitch error (according to GPE) or a voicing decision error (according to VDE).

In addition to these metrics, we propose a subjective (i.e., human) test structured as an AXY discrimination test that we refer to as an “anchored prosody side-by-side”. A human rater is presented with three stimuli: a reference speech sample (A), and two competing samples (X and Y) to evaluate. The rater is asked to rate whether the prosody of X or Y is closer to that of the reference on a 7-point scale. The

Table 1. A summary of quantitative and subjective metrics (Section 4.2) used to evaluate the prosody transfer. Lower is better for both MCD$_k$ and FFE. Higher subjective scores are better, and indicate whether human raters believe the voice is closer in prosody to the reference than the corresponding baseline model on a 7 point (-3 to 3) scale, where 0 is "about the same".

VOICE	MODEL	REFERENCE	MCD$_{13}$	FFE	SUBJECTIVE
SINGLE-SPEAKER	BASELINE	SAME SPEAKER	10.63	53.2%	
SINGLE-SPEAKER	TANH-128	SAME SPEAKER	**7.92**	**28.1%**	$\mathbf{1.611 \pm 0.164}$
SINGLE-SPEAKER	BASELINE	UNSEEN SPEAKER	11.22	59.6%	
SINGLE-SPEAKER	TANH-128	UNSEEN SPEAKER	**8.89**	**38.0%**	$\mathbf{1.465 \pm 0.132}$
MULTI-SPEAKER	BASELINE	SAME SPEAKER	9.93	48.5%	
MULTI-SPEAKER	TANH-128	SAME SPEAKER	**6.99**	**27.5%**	$\mathbf{1.307 \pm 0.127}$
MULTI-SPEAKER	BASELINE	SEEN SPEAKER	12.37	64.2%	
MULTI-SPEAKER	TANH-128	SEEN SPEAKER	**9.51**	**37.1%**	$\mathbf{0.871 \pm 0.138}$
MULTI-SPEAKER	BASELINE	UNSEEN SPEAKER	11.84	60.0%	
MULTI-SPEAKER	TANH-128	UNSEEN SPEAKER	**10.87**	**41.3%**	$\mathbf{1.146 \pm 0.246}$

scale ranges from "X is much closer" to "Both are about the same distance" to "Y is much closer", and can naturally be mapped on the integers from -3 to 3. Prior to collecting any ratings, we provide the raters with 4 examples of prosodic attributes to evaluate (intonation, stress, speaking rate, and pauses), and explicitly instruct the raters to ignore audio quality or pronunciation differences. A screenshot of this user interface is included in the supplemental material. For each triplet (A, X, Y) evaluated, we collect 4 independent ratings. No rater is used for more than 6 items in a single evaluation. To analyze the data from these subjective tests, we average the scores and compute 95% confidence intervals.

4.3. Same-text Prosody Transfer

We first demonstrate that our model is capable of prosody transfer when the text is unchanged from that of the reference utterance.

4.3.1. SPECTROGRAMS AND PITCH TRACKS

Figure 4 shows three spectrograms (reference, baseline model, prosody embedding model) for the same utterance. Note that the spectrogram from the model conditioned on a reference embedding bears a much stronger resemblance to the reference signal than that generated by an unconditioned model. In particular, notice that the spectrogram from the baseline model, which does not use a reference signal, exhibits noticeably different rhythm – for example, there is a long pause between the two halves of the utterance, and the utterance lasts much longer. By contrast, the output with a prosody embedding has the same length and pause characteristics as the reference audio; it also has recognizably similar harmonic and onset structure.

Figure 5 shows the pitch tracks for the same triplet of utterances. We can see that the prosody embedding model closely follows the pitch contours of the reference, whereas the unconditioned model does something else entirely.

4.3.2. QUANTITATIVE AND SUBJECTIVE EVALUATIONS

We evaluated synthesis of single- and multi-speaker models using two types of reference utterance. "Same speaker" indicates a reference utterance from the same speaker as the target, while "unseen speaker" refers to a reference utterance from a speaker unseen in training. For the multi-speaker model, we also tested synthesis with a speaker seen in training but different from the target speaker ("seen speaker").

We present our findings in Table 1. The results show that augmenting Tacotron with a reference encoder allows it to match the reference prosody substantially more accurately. This is true for all baseline/model pairs in Table 1, and is independent of whether the reference speaker matches the target speaker. The objective metrics MCD$_{13}$ and FFE also support this conclusion, both resulting in substantially lower values for the reference encoder model than for the baseline model.

Note that when the target and reference speakers are different (i.e., when the reference in Table 1 is either "seen speaker" or "unseen speaker"), we must be careful to demonstrate that prosody transfer has been achieved. If the bottleneck allows too much information to flow through the reference encoder, for example, the overall model could simply copy the reference to the output. In this instance, listening to even a small number of outputs suffices to verify that the output speaker matches the target speaker, and that we have in fact achieved prosody transfer across speakers. However, further experiments, explored in Section 4.5, provide some surprising results.

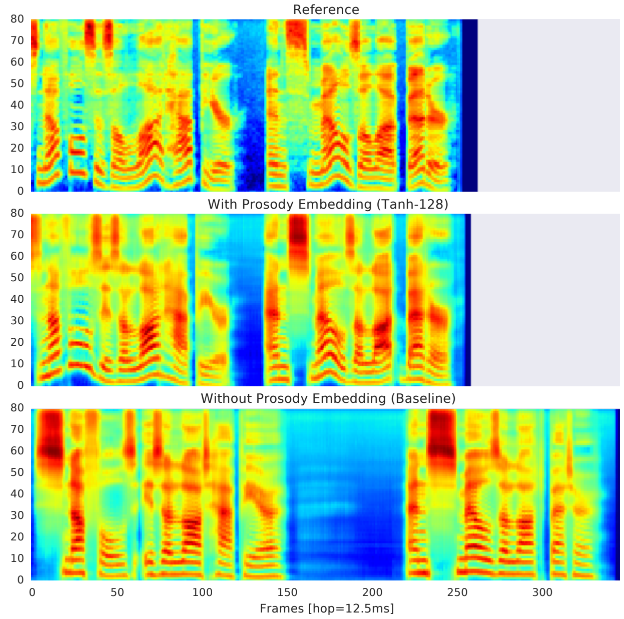

Figure 4. Mel spectrograms for the utterance “Snuffles is a lot happier. And smells a lot better.” (Top) Reference utterance from an unseen speaker. (Middle) Synthesized utterance conditioned on reference embedding. (Bottom) Synthesized utterance from a model without reference conditioning.

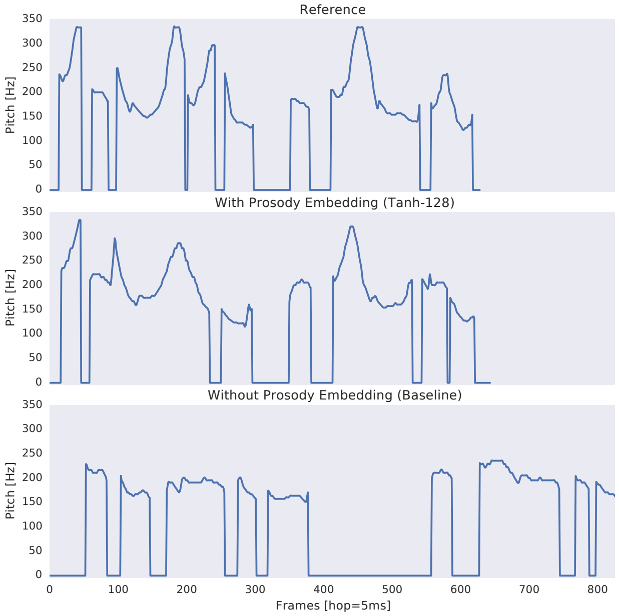

Figure 5. Pitch tracks for the utterance “Snuffles is a lot happier. And smells a lot better.” A pitch of 0 Hz indicates an unvoiced segment. (Top) Reference utterance from an unseen speaker. (Middle) Synthesized utterance conditioned on reference embedding. (Bottom) Synthesized utterance from a model without reference conditioning.

4.4. Templated Prosody Transfer

In addition to same-text prosody transfer, we also explore the robustness of our proposed model to changes in the synthesized text. Since the prosody embeddings we learn capture prosodic features with some fine time detail, it isn’t clear what it would mean to transfer these prosodic features to a radically different utterance. As expected, we find that drastic changes to the sentence or phrase structure result in undesirable prosody transfer. This use case may be more suited to models that capture less granular features of prosody such as emotion or style. (Wang et al., 2018), for example, applies a similar approach to learning representations of global style.

Nonetheless, we include a number of examples on our demo page demonstrating that text transformations can be performed without compromising intelligibility or desired prosody. This can be highly useful in building templated dialogue systems capable of synthesizing a template with a desired prosody.

4.5. Preservation of Speaker Identity

In Table 1, the results of our “anchored prosody side-by-side” subjective evaluation show that reference-based synthesis matches the reference audio significantly better than the baseline model. However, the evaluation does not assess whether the target speaker identity was preserved by the synthesis. This is not accidental: pitch, pacing, and other prosodic characteristics factor into speaker identity, and thus it is difficult to prescribe exactly which aspects of the target speaker’s identity should be preserved during prosody transfer.

The audio samples we include on our demo page show that our model preserves many important aspects of speaker identity during prosody transfer. We include a grid of audio examples representative of typical performance of this system, with reference clips from 6 speakers with distinct accents. Each utterance is synthesized 6 times, each with a different target speaker. Notably, the prosody of each clip matches that of the reference, while the distinct accents and vocal tract properties of each speaker are preserved.

However, listening to samples of a male voice controlling a female voice (and vice-versa) reveals that our prosody representation encodes pitch in an absolute manner. When controlled by a male reference signal, female target speakers sound as if they’re imitating a person with a deeper voice. Similarly, when controlled by a female reference signal, male speakers sound as if they’re imitating a person with a higher voice. This suggests that the prosody and speaker representations are somewhat entangled.

To quantify this entanglement, we designed a simple speaker identification model that takes varying types of acoustic input, and produces predictions of speaker identity from a universe of speakers known at training time. The architecture uses the same strided convolutions and GRU-based aggregation as the reference encoder architecture from Section 3.2, and is independently trained on ground truth mel spectrograms using the same 44-speaker dataset used to train our multi-speaker model. The architecture achieves over 99% accuracy on both the held-out ground truth and synthesized audio from our baseline 44-speaker model.

We then tested our prosody-enhanced Tacotron using this model. To do so, we first constructed pairs of all target speakers and reference utterances in the test set. We then used our prosody-enhanced Tacotron to generate mel spectrograms for these pairs, and fed the output into the speaker identification model. The speaker identification model identified the spectrograms as originating from the reference speaker in 61% of test set examples, and the target speaker only 21% of the time (ideally, the target speaker would be at 100%). We refer the reader to the audio samples to understand how surprising this is – the audio samples *sound* substantially more like the target speaker in every sample we've listened to.

Since our model seems to transfer prosody in a pitch-absolute manner, we ran a further experiment where we trained the speaker identification model on 13 mel-frequency cepstral coefficients (MFCCs) which contain less pitch content. In this case, the speaker identification model identified the utterances as originating from the reference speaker 41% of the time, and the target speaker 32% of the time, suggesting that, indeed, speaker-dependent pitch content is transferred from the reference to the output.

4.6. Bottleneck Size and Shape

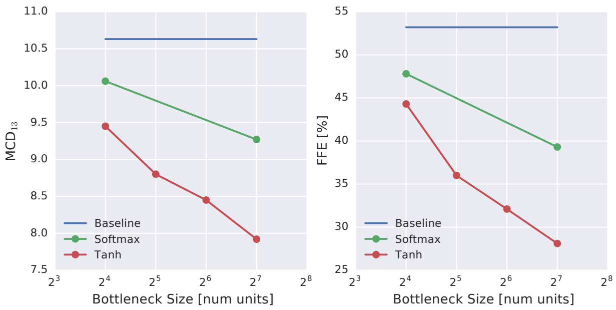

Figure 6. The effect of bottleneck size on quantitative metrics. In terms of both MCD_{13} and FFE, models with prosody encoders beat the baseline. As the bottleneck size increases, the performance in both metrics improve. Softmax is a more severe bottleneck than tanh, and exhibits worse metrics.

The dimensionality and activation used for the bottleneck substantially affect the information flow from the prosody reference encoder to the output. In this experiment, we use our single speaker as both the reference signal and target (we are essentially trying to conditionally autoencode the mel spectrograms given text). We plot the MCD_{13} and FFE metrics while varying the bottleneck size and activation in Figure 6, and include a series of audio samples on our demo page. We can conclude that increasing the bottleneck size allows for significantly more information flow from the reference to the output, allowing for better reproduction of the reference. More interestingly, using a softmax activation leads to a degradation of metrics in comparison to tanh: this is probably due to the exponential suppression of the non-maximal components in the softmax.

The quantitative metrics are in agreement with the audio samples provided on our demo page: larger bottlenecks with the tanh activation improve audio similarity, and the outputs are more faithful to the reference prosody. A potential tradeoff is that a narrower bottleneck would likely better preserve the speaker identity of the target speaker.

5. Discussion and Future Work

In this work, we have demonstrated prosody transfer via an end-to-end learned representation of prosody directly from acoustic signals. While our system successfully transfers prosody from one speaker to another, it does so in a pitch-absolute manner. Future work should focus on encoding prosody in a pitch-relative manner so that speaker identity is more completely preserved during transfer.

A substantial open question is how to disentangle the textual information implicit in the reference signal from the prosodic information. In Section 4.4, we showed that this is possible to some extent, especially when the transcripts are relatively close. But, more generally, this amounts to transferring or controlling prosody using utterances with different corresponding text transcripts. As noted earlier, this is a somewhat ill-defined task, and a more careful formalization of this problem is needed to make real progress.

We also defined objective and subjective metrics for evaluating prosody transfer, and evaluated our architecture on these benchmarks. Solidifying metrics that quantify all desired aspects of prosody transfer (e.g., prosodic similarity and the degree to which prosodic, textual, and speaker information are disentangled) is an important step in the long-term progression of end-to-end prosody work.

Finally, given our construction of a prosody space, we would like to be able to sample from this space (i.e., generate prosody instead of transferring it). One could, for example, attempt to learn a prior distribution over the prosody space.

Acknowledgements

The authors thank Aren Jansen and the Machine Hearing, Google Brain, and Google TTS teams for their helpful discussions and feedback.

References

Arık, S. O., Diamos, G., Gibiansky, A., Miller, J., Peng, K., Ping, W., Raiman, J., and Zhou, Y. Deep voice 2: Multi-speaker neural text-to-speech. *arXiv preprint arXiv:1705.08947*, 2017.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014a. URL https://arxiv.org/abs/1406.1078.

Cho, K., van Merrienboer, B., Gulcehre, C., Bougares, F., Schwenk, H., and Bengio, Y. *Learning phrase representations using RNN encoder-decoder for statistical machine translation*. 2014b.

Chorowski, J., Weiss, R. J., Saurous, R. A., and Bengio, S. On using backpropagation for speech texture generation and voice conversion. *arXiv preprint arXiv:1712.08363*, 2017.

Chu, W. and Alwan, A. Reducing f0 frame error of f0 tracking algorithms under noisy conditions with an unvoiced/voiced classification frontend. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 3969–3972. IEEE, 2009.

Coile, B. V., Tichelen, L. V., Vorstermans, A., Jang, J. W., and Staessen, M. PROTRAN: a prosody transplantation tool for text-to-speech applications. In *The 3rd International Conference on Spoken Language Processing, ICSLP 1994, Yokohama, Japan, September 18-22, 1994*, 1994. URL http://www.isca-speech.org/archive/icslp_1994/i94_0423.html.

De Cheveigné, A. and Kawahara, H. Yin, a fundamental frequency estimator for speech and music. *The Journal of the Acoustical Society of America*, 111(4):1917–1930, 2002.

Eyben, F., Buchholz, S., and Braunschweiler, N. Unsupervised clustering of emotion and voice styles for expressive tts. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pp. 4009–4012. IEEE, 2012.

Glorot, X. and Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256, 2010.

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

Griffin, D. and Lim, J. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(2):236–243, 1984.

Hirst, D. Automatic analysis of prosody for multilingual speech corpora. *Improvements in speech synthesis*, pp. 320–327, 2001.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pp. 448–456, 2015.

Kingma, D. and Ba, J. Adam: A method for stochastic optimization. *International Conference for Learning Representations*, 2015.

Kinnunen, T., Juvela, L., Alku, P., and Yamagishi, J. Non-parallel voice conversion using i-vector plda: Towards unifying speaker verification and transformation. In *ICASSP*, 2017.

Kubichek, R. Mel-cepstral distance measure for objective speech quality assessment. In *Communications, Computers and Signal Processing, 1993., IEEE Pacific Rim Conference on*, volume 1, pp. 125–128. IEEE, 1993.

Ladd, D. R. *Intonational phonology*. Cambridge University Press, 2008.

Nakashika, T., Takiguchi, T., Minami, Y., Nakashika, T., Takiguchi, T., and Minami, Y. Non-parallel training in voice conversion using an adaptive restricted boltzmann machine. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 24(11):2032–2045, November 2016.

Nakatani, T., Amano, S., Irino, T., Ishizuka, K., and Kondo, T. A method for fundamental frequency estimation and voicing decision: Application to infant utterances recorded in real acoustical environments. *Speech Communication*, 50(3):203–214, 2008.

Nose, T., Yamagishi, J., Masuko, T., and Kobayashi, T. A style control technique for hmm-based expressive speech synthesis. *IEICE TRANSACTIONS on Information and Systems*, 90(9):1406–1413, 2007.

Obin, N., Beliao, J., Veaux, C., and Lacheret, A. SLAM: Automatic stylization and labelling of speech melody. In *Speech Prosody*, pp. 246–250, 2014.

Rosenberg, A. AuToBI-a tool for automatic ToBI annotation. In *Interspeech*, pp. 146–149, 2010. URL http://eniac.cs.qc.cuny.edu/andrew/autobi/.

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerry-Ryan, R., Saurous, R. A., Agiomyrgiannakis, Y., and Wu, Y. Natural tTS synthesis by conditioning wavenet on mel spectrogram predictions. *arXiv preprint arXiv:1712.05884*, dec 2017. URL https://arxiv.org/abs/1712.05884.

Silverman, K., Beckman, M., Pitrelli, J., Ostendorf, M., Wightman, C., Price, P., Pierrehumbert, J., and Hirschberg, J. ToBI: A standard for labeling english prosody. In *Second International Conference on Spoken Language Processing*, 1992.

Stevens, S. S., Volkmann, J., and Newman, E. B. A scale for the measurement of the psychological magnitude pitch. *The Journal of the Acoustical Society of America*, 8(3): 185–190, 1937.

Taylor, P. The tilt intonation model. 1998.

Tesser, F., Sommavilla, G., Paci, G., and Cosi, P. Experiments with signal-driven symbolic prosody for statistical parametric speech synthesis. In *Eighth ISCA Workshop on Speech Synthesis*, 2013. URL http://ssw8.talp.cat/papers/ssw8_PS2-7_Tesser.pdf.

van den Oord, A., Vinyals, O., et al. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, pp. 6309–6318, 2017.

Wagner, M. and Watson, D. G. Experimental and theoretical advances in prosody: A review. *Language and cognitive processes*, 25(7-9):905–945, 2010.

Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., Le, Q., Agiomyrgiannakis, Y., Clark, R., and Saurous, R. A. Tacotron: Towards end-to-end speech synthesis. In *Proceedings of Interspeech*, August 2017a. URL https://arxiv.org/abs/1703.10135.

Wang, Y., Skerry-Ryan, R., Xiao, Y., Stanton, D., Shor, J., Battenberg, E., Clark, R., and Saurous, R. A. Uncovering latent style factors for expressive speech synthesis. *ML4Audio Workshop, NIPS*, 2017b.

Wang, Y., Stanton, D., Zhang, Y., Skerry-Ryan, R., Battenberg, E., Shor, J., Xiao, Y., Ren, F., Jia, Y., and Saurous, R. A. Style Tokens: Unsupervised Style Modeling, Control, and Transfer in End-to-End Speech Synthesis. *International Conference on Machine Learning*, 2018. URL https://arxiv.org/abs/1803.09017.

Wightman, C. W. ToBI or not ToBI? In *Speech Prosody 2002, International Conference*, 2002.

Wu, Z., Chng, E. S., and Li, H. Conditional restricted boltzmann machine for voice conversion. In *ChinaSIP*, 2013.