CS230

Accent Transfer with Discrete Representation Learning and Latent Space Disentanglement

Renee Li
Dept. of Computer Science
Stanford University
reneeli@stanford.edu

Paul Mure
Dept. of Computer Science
Stanford University
paulmure@stanford.edu

Abstract

The task of accent transfer has recently become a field actively under research. In fact, several companies were established with the goal of tackling this challenging problem. We designed and implemented a model using WaveNet and a multitask learner to transfer accents within a paragraph of English speech. In contrast to the actual WaveNet, our architecture made several adaptations to compensate for the fact that WaveNet does not work well for drastic dimensionality reduction due to the use of residual and skip connections, and we added a multitask learner in order to learn the style of the accent.

1 Introduction

In this paper, we explored the task of accent transfer in English speech. This is an interesting problem because it shows the ability of neural network to generate novel contents based on partial information, and this problem has not been explored as much as some of the other topics in style transfer. We believe that this task is very interesting and challenging because we will be developing the state-of-the-art model for our specific task: while there are some published papers about accent classification, so far, we have not been able to find a good pre-trained model to use for evaluating our results. Moreover, raw audio generative models are also a very active field of research, so there is not much literature or implementations for relevant tasks. This task also has significant social implications as it has been shown that accent can impact one's salary in the workforce.

2 Related work

Van den Oord et al. at Google DeepMind showed the incredible potential for Vector Quantised-Variational AutoEncoder (VQ-VAE) for neural discrete representation learning [2]. In their paper, they mentioned the application for VQ-VAE on audio style transfer with promising results. Furthermore, models built on top of autoencoders have shown great potential in style transfer as applied to text [3], [4]. This usually involves adding one or more adversaries and multitask learners on top of the encoder to facilitate the latent space disentanglement with respect to the style and content space.

Another paper that gave us inspiration is the original WaveNet paper [7]. WaveNet is a deep neural network for generating raw audio waveforms. The model is "fully probabilistic and autoregressive, with the predictive distribution for each audio sample conditioned on all previous ones" [7]. It is known to be particularly fitting for audio-related tasks. For example, when applied to text-to-speech, WaveNet yields state-of-the-art performance. The authors also showed that it can be employed as a discriminative model, returning promising results for phoneme recognition, which gave us inspiration to use it for our task at hand.

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

3 Dataset and Features

We used the dataset from the Speech Accent Archive, which consists of parallel English speech samples from 177 countries. There are 2,138 speech clips of the same passage in English in 177 different accents. We used 1,710 randomly selected speech clips for our training set, 214 speech clips for our validation set, and 214 speech clips for our test set. The passage that all of the participants read was:

> "Please call Stella. Ask her to bring these things with her from the store: Six spoons of fresh snow peas, five thick slabs of blue cheese, and maybe a snack for her brother Bob. We also need a small plastic snake and a big toy frog for the kids. She can scoop these things into three red bags, and we will go meet her Wednesday at the train station."

The raw input to our algorithm are audio clips of English speeches with different accents in .mp3 format; during data pre-processing, we converted them into .wav format so that we can feed them into the WaveNet. Each audio clip was then partitioned into two-second clips sampled at 16384 Hz. Then, these two-second clips are normalized to take on real values between -1 and 1. After all the preprocessing, our dataset totaled to 19,450 distinct 2 seconds audio clips with 207 different accent labels. The output of the model is now raw audio waveform that can be easily converted into .wav format as our final output.

4 Approach

We planed on implementing a vector quantised variational autoencoder (VQ-VAE) in combination with a multitask classifier and an adversary to disentangle the content information from the style information in the latent space. This encoder seeks to reduce the dimensionality of the input feature by extracting the high level features in the latent space, then a decoder is trained to reconstruct the original input from that high level overview. More specifically, the encoder would output a 256 dimension vector where the first 32 numbers represent a style encoding and the other 224 representing content information. In order to facilitate the disentanglement, we have a multi-layer perceptron classifying the style label of the style space that is trained with the encoder. In our original model, we had another classifier as an adversary and tries to classify the style label of the content space, and we punish the encoder if the adversary succeeds. Unlike the style classifier, the adversary's back propagation does not extend to the encoder. However, we found out later that the adversary does not work well with our model because its gradient explodes and overshadows the other losses which are more important. We have tried scaling it down and gradient clipping, neither approach produced satisfying results, so we decided to remove the adversary from our model for the scope of this project.

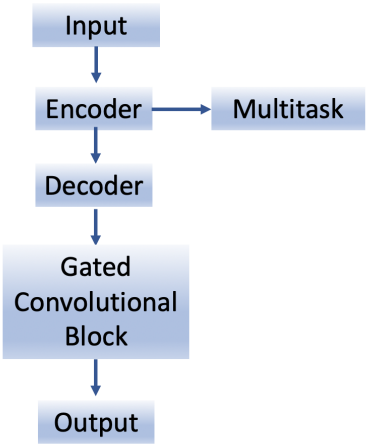


Figure 1: Our model architecture.

In our final model (see Figure 1), the inputs are raw audio clips sampled at 16384 Hz; each data entry is a two-second audio clip, so the input dimension is 1 X 32768. The encoder is an existing Wavenet implementation that takes in raw audio inputs and produces a 256 X 1 encoding. The encoder itself is composed 2 Wavenet blocks, and each block has 14 layers, with a structure shown in Figure 2.

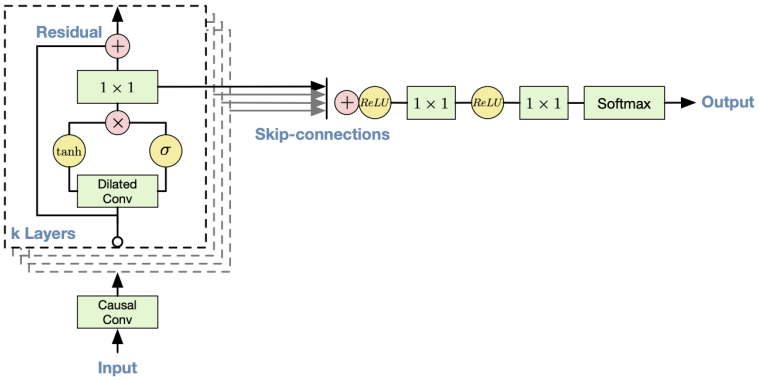


Figure 2: Structure of each block within the encoder and decoder.

Then the encoding is quantized through a vector quantizer and fed into the decoder. For the decoder, we used the pre-exisiting implementation of the $wavenet\_vocoder$ (with 2 blocks and 10 layers per block) that takes in the encoded latent vector and produces a 128 X 256 output. For our final layer, after the $wavenet\_vocoder$ spits out the 128 X 256 output, there is one more gated convolutional block with 7 layers that transposes the 128 X 256 result back up to the 1 X 32768 dimension of the input audio. No residual or skip connection is used in this final block due to the upsampling involved in the transposition. In total, the decoder has 27 gated convolutional layers as described in the WaveNet paper. As for the multitask in our final model, we chose to use a simple one-layer dense network with a softmax activation that takes the first 32 numbers of the latent space and predicts the style label.

Some of the hyperparameters that we have tried to tune include the architecture of the encoder used, whether it uses a WaveNet with raw audio as input, or if it take in MFCCs (explained in Section 6) as input, the number of convolutional layers (7 to 28) that the encoder uses, number of gates and filters associated with the decoder, whether the encoder and/or decoder utilizes residual connection, different sizes of the latent space dimensions (which can be anywhere from 64 to 512), different number of discrete embeddings in the vector quantizer codebook (which can be anywhere from 64 to 1024), and different sizes of multitask and adversary (between one and four layers).

After experimenting with many different hyperparameters, we found that the model with the vector quantizer always converge to producing silence within a few epochs (less than 5 in most cases), so in our final model, we decided to remove the vector quantizer, which made the model converge much slower (usually after 15 epochs). The initial output of the model without the vector quantizer were also better than the one with the quantizer.

4.1 Loss Function

There are two main parts to our final loss function: $L_{reconstruction}$ and $L_{multitask}$, each defined below:

$$L_{multitask} = -\Sigma_{l \in labels} t_s(l) log(y_s(l))$$

$$L_{reconstruction} = ||x - \hat{x}||_2^2$$

Our final loss is defined as $L_{final} = L_{reconstruction} + \lambda L_{multitask}$.
Where λ is a constant hyperparameter that needs to be tuned.

5 Evaluation

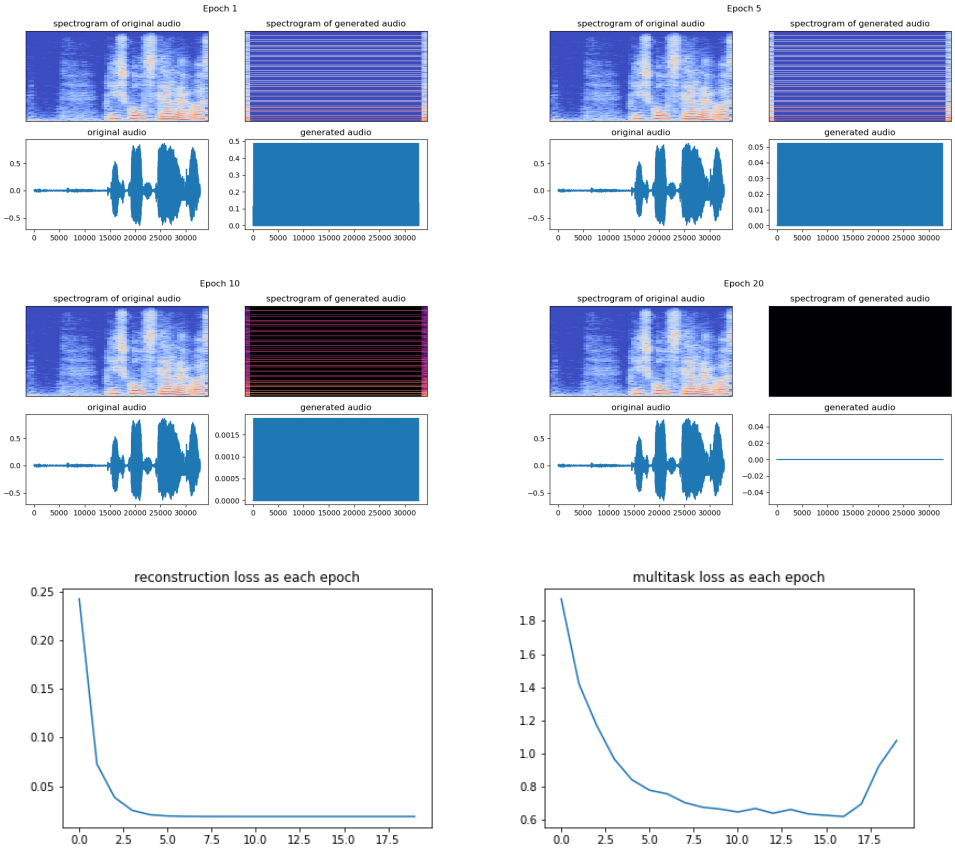
We expect our model to take in voice recordings and output voice recordings of the same content in a different accent. Therefore, we can evaluate the style transfer accuracy and the content preservation with additional pre-trained networks in audio accent classification and transcription. Both transfer accuracy and content preservation will be measured as $\frac{N_{correct}}{N_{total}}$, where $N_{correct}$ is the number of correctly labeled results, and N_{total} is the total size of our output data.

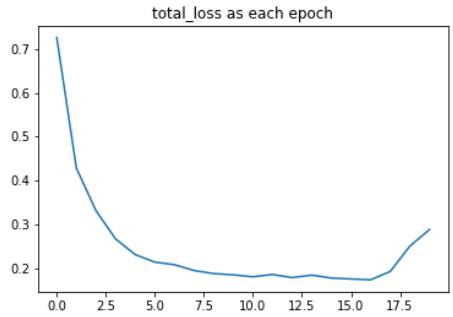
6 Results and Analysis

As mentioned above, initially, we experimented with models that took in Mel-frequency cepstral coefficients (MFCCs) of the original audio that tried to synthesize the raw audio out of that. The idea is that MFCCs would have encoded meaningful timbral and textual information about that audio that would ease the work of the encoder, thus requiring a much smaller network. However, the models we tried always converge within a couple of epochs and produce nothing but silence with a bump in the beginning and end.

Subsequently, we drastically increased the size of our model by using a WaveNet at both the encoder and decoder. Intially, this model seems to be yielding resulsts that look a lot more like actual audio waveforms, but the output is still just pure noise. And surprisingly, the models always converge towards silence. As mentioned before, removing the vector quantizer seems to make the model converge more slowly, but eventually the model still converged towards silence after 17 epochs, as can be seen in the spectrograms demonstrated below.

With the size of our final model and dataset, the computation cost already exceeded the scope of a 10 week class project, since the model was takes hours to train one epoch. It is very possible that going bigger, both in terms of the size of the model and the size of the dataset will yield more promising results. But as we have shown, the task is likely not doable with the size of our model.





One hypothesis for the noisy output comes from the nature of how we are training the multitask. Our multitask predicts an accent label, whereas the encoder-decoder network deals exclusively with synthesizing audio. Therefore, one hypothesis for the noisy output of the model is that during backpropagation, the parameters from the multitask module ware polluting the encoder parameters, thus causing the output to have unexpected results. Furthermore, if that is indeed the case, and we are not actually constrained by the size of our model, then we can conclude that the fundamental idea of latent space disentanglement with a multitask and adversary as we know it might not work with raw audio.

7 Conclusion and Future Work

Our model architecture requires that the model learn a concise encoding from an audio clip with over 30,000 time steps, which naturally requires a sizable model to do so. This, coupled with the fact that some portion of the latent space must go to encoding style information, makes the task especially difficult. Although this idea of latent space disentanglement produced promising results in applications to text data, it will be difficult to apply it in synthesizing raw audio.

While we have been able to come up with an initial implementation based on WaveNet we can potentially incorporate the $JITTER$ layer of the WaveNet structure, which is a relatively novel concept that we would like to explore [6]. $JITTER$ have the potential to make the model more robust because it lowers the model's dependence on the consistency across groups of token. During training, $JITTER$ allows each latent vector to replace either one or both of its neighbors. This regularization also promotes latent representation stability over time. On the other hand, while doing this project, we realized that there is a lack of good and large enough datasets that can be used for this task. In fact, one of main challenges for this task is keeping a consistent performance across different "target accents" given an imbalanced dataset because the amount of data (speech clips). Thus, a valuable potential future work is to develop larger and more balanced dataset of audio clips with different accents in English speech.

8 Code

https://github.com/paulmure/AccentTransfer

9 Contributions

Both Paul Mure and Renee Li conducted background research for this project. Paul Mure implemented the models. Renee Li wrote most of the report. Both authors contributed equally.

References

[1] Tatman, R. (2017). Speech Accent Archive. Kaggle, www.kaggle.com/rtatman/speech-accent-archive/metadata.

[2] Van Den Oord, A., & Vinyals, O. (2017). Neural discrete representation learning. In Advances in Neural Information Processing Systems (pp. 6306-6315).

[3] John, V., Mou, L., Bahuleyan, H., & Vechtomova, O. (2018). Disentangled representation learning for non-parallel text style transfer. arXiv preprint arXiv:1808.04339.

[4] Fu, Z., Tan, X., Peng, N., Zhao, D., & Yan, R. (2017). Style transfer in text: Exploration and evaluation. arXiv preprint arXiv:1711.06861.

[5] Chen, L., Shen, L., & Tang, M. (2018). Accent Classification and Neural Accent Transfer of English Speech. CS230: Deep Learning, Winter 2018, Stanford University.

[6] Chorowski, J., Weiss, R., Bengio, S., & Oord, A. (2019). Unsupervised Speech Representation Learning Using WaveNet Autoencoders. IEEE/ACM Transactions on Audio, Speech, and Language Processing. PP. 1-1. 10.1109/TASLP.2019.2938863.

[7] Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., ... & Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. arXiv 2016. arXiv preprint arXiv:1609.03499.