pISSN 2005-8063
eISSN 2586-5854
2021. 12. 31.
Vol.13 No.4
pp. 47-53

말소리와 음성과학

Phonetics and Speech Sciences

한국음성학회지

https://doi.org/10.13064/KSSS.2021.13.4.047

End-to-end non-autoregressive fast text-to-speech

Wiback Kim · Hosung Nam*

Department of English Language and Literature, Korea University, Seoul, Korea

Abstract

Autoregressive Text-to-Speech (TTS) models suffer from inference instability and slow inference speed. Inference instability occurs when a poorly predicted sample at time step t affects all the subsequent predictions. Slow inference speed arises from a model structure that forces the predicted samples from time steps 1 to t-1 to predict the sample at time step t. In this study, an end-to-end non-autoregressive fast text-to-speech model is suggested as a solution to these problems. The results of this study show that this model's Mean Opinion Score (MOS) is close to that of Tacotron 2 - WaveNet, while this model's inference speed and stability are higher than those of Tacotron 2 - WaveNet. Further, this study aims to offer insight into the improvement of non-autoregressive models.

Keywords: deep learning, neural network, speech synthesis, Text-to-Speech (TTS)

1. 서론

TTS(text-to-speech)는 주어진 문자열을 음성으로 출력하는 모든 시스템을 의미한다. Yarrington(2007)은 TTS를 세 종류로 구별한다. Articulatory synthesis는 인간의 성도(vocal tract)를 모방한 조음 모델에서 음성을 출력한다. Formant synthesis는 음소별로 formant를 조정해서 음성을 출력한다. Concatenative synthesis는 사전에 녹음된 음소를 결합해서 음성을 출력한다. 근래에는 인공신경망(artificial neural network)의 발전에 힘입어 각양각색의 인공신경망을 이용한 TTS가 등장하고 있다.

보통의 인공신경망 TTS는 autoregressive하다는 특성을 가진다. 순환신경망(recurrent neural network, RNN)이 사용되기 때문이다. 여기서 순환신경망은 시계열 데이터를 처리하는 인공신경망이고, 시계열 데이터는 시간에 따라 순서대로 나열된 데이터를 말한다. TTS는 시계열 데이터인 음성을 출력하기 때문에 순환신경망을 이용하고, 순환신경망은 autoregressive한 특징을 지니므로 보통의 TTS는 autroregressive하다고 볼 수 있는 것이다. Autoregression은 time step 1부터 t-1까지의 데이터를 참고해서 time step t의 데이터를 출력하는 행위이다.

Autoregressive TTS에 대한 선행 연구는 대부분 Google에서 행해졌다. Google의 Tacotron(Wang et al., 2017)은 RNN encoder, RNN decoder, CBHG 모듈, Griffin-Lim algorithm(Griffin & Lim, 1983)으로 음성을 출력한다. WaveNet(van den Oord et al., 2016)은 합성곱신경망(convolutional neural network)을 이용하지만 inference를 할 때 autoregressive한 원칙을 고수하면서 음성을 출력한다. Tacotron 2(Shen et al., 2017)는 CBHG 모듈 없이도 성능

* hnam@korea.ac.kr, Corresponding author

Received 1 August 2021; Revised 28 September 2021; Accepted 4 October 2021

이 좋고 WaveNet과 합쳐질 수 있다. 이밖에도 Tacotron을 normalizing flow에 맞게 개조한 NVIDIA의 Flowtron(Valle et al., 2020), 음소열 변환, 길이 예측, f0 예측, 음성 출력을 하는 모델 4개로 제작된 Baidu의 Deep Voice(Arik et al., 2017) 등이 있다.

Autoregressive TTS는 output을 통째로 출력하지 못하기 때문에 출력 속도가 느리다는 결함이 있다(Wang et al., 2020). Time step t의 데이터를 출력하기 위해서는 time step 1부터 *t*-1까지의 데이터를 순서대로 출력해야 하기 때문이다. Time step 1의 데이터를 잘못 예측하면 이후의 모든 time step에 대한 데이터도 잘못 예측할 불안정성도 있다.

Non-autoregressive TTS는 autoregressive TTS의 속도 문제에의 대안으로 나온 것들이 많으며 autoregressive한 특성의 순환신경망 대신 합성곱신경망을 주로 사용한다. 문자열에서 mel-scale spectrogram을 출력하는 non-autoregressive 모델인 Microsoft와 Zhejiang University의 FastSpeech(Ren et al., 2019)는 teacher 모델로부터 attention을 배운다.

Mel-scale spectrogram에서 음성을 출력하는 non-autoregressive 모델은 종류가 더 많다. Google의 Parallel WaveNet(van den Oord et al., 2017)은 학습이 잘 된 teacher WaveNet으로부터 확률 분포를 배우고 나서 음성을 더 빠르게 출력한다. Lyrebird와 Montreal University의 MelGAN(Kumar et al., 2019)과 NAVER의 Parallel WaveGAN(Yamamoto et al., 2019)은 생성적 적대 신경망(generative adversarial network)으로 제작된 모델이다.

상기한 non-autoregressive TTS는 출력 속도가 빠르지만 모델 2개를 각자 훈련 후 합쳐야 하는 번거로움이 있다. FastSpeech 등은 mel-scale spectrogram을 출력하기 때문에 뒷단에 음성 출력 모델이 있어야 한다. Parallel WaveNet, MelGAN, Parallel WaveGAN 등은 전자 모델의 뒷단에 연결돼서 음성을 출력한다.

본 연구는 autoregressive TTS의 출력 속도 및 안정성 문제가 없고 모델 1개로 제작되는 end-to-end non-autoregressive TTS를 목표로 했다. 이에 따라 본 연구는 FastSpeech(Ren et al., 2019)와 DCTTS(Tachibana et al., 2017)를 참고한 non-autoregressive TTS를 제안하고, autoregressive TTS와 비교해서 모델을 평가했다.

2. 모델

그림 1은 본 연구가 제안하는 모델의 구성도이다.

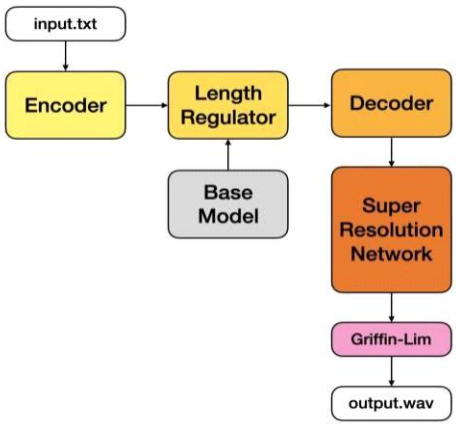

그림 1. 본 연구의 모델
Figure 1. Non-autoregressive model used in this paper

모델은 네 부분으로 분할 가능하다. Encoder, length regulator, decoder는 Transformer(Vaswani et al., 2017)와 FastSpeech(Ren et al., 2019)를 참조했다. Super-resolution network는 DCTTS (Tachibana et al., 2017)를 참조했다.

2.1. Encoder와 decoder

2.1.1. Encoder와 decoder

Encoder와 decoder는 많은 인공신경망 모델에서 이용하는 구조이다. Encoder는 input에서 중요한 특징을 출력한다. Decoder는 encoder가 출력한 특징을 input으로 하여 output을 출력한다. 이 구조는 대체로 데이터를 다른 특성의 데이터로 바꿀 때 이용된다. 예를 들면, Autoencoder는 input과 동일한 형태의 output을 출력하도록 설계된 모델로 이미지의 노이즈 제거 작업을 수행할 수 있다(Cho, 2013). Encoder가 노이즈가 들어간 input을 받고, decoder가 노이즈가 제거된 input을 출력하도록 학습시키면 된다.

다른 예시로는 Seq2seq(Sutskever et al., 2014)이 있다. Seq2seq은 sequence-to-sequence의 약어로, 이름 그대로 순차 데이터를 다른 특성의 순차 데이터로 바꾸는 encoder와 decoder가 결합된 모델이다. 순차 데이터는 데이터를 형성하는 샘플의 순서가 중시되는 데이터로 문자열, 음성 특징 등이 있다. Seq2seq의 encoder와 decoder는 순차 데이터를 처리하고자 순환신경망으로 조직되며(Sutskever et al., 2014), encoder RNN과 decoder RNN으로 통칭되기도 한다. Seq2seq은 input 언어를 다른 언어로 바꾸는 번역기, input 문장에 대한 답변을 출력하는 챗봇, 문자열을 음성으로 바꾸는 TTS 등 많은 영역에서 응용된다(Shen et al., 2017; Sutskever et al., 2014).

그러나 순환신경망으로 조직된 Seq2seq은 상기했던 대로 autoregressive하다는 특징 때문에 속도 저하 및 불안정성 문제를 가진다. 본 연구가 제안하는 모델은 autoregression을 피하기 위해 FastSpeech처럼 Transformer에 기초해서 encoder와 decoder를 제작했다. 본 연구의 encoder와 decoder는 합성곱신경망 3개, 심층신경망(deep neural network) 1개와 Transformer의 Multi-Head

Self-Attention 5개를 사용했으며, head는 8개로 설정했다. 이를 한 개의 stack으로 본다면 encoder와 decoder는 각각 6개의 stack을 사용했다.

2.1.2. Transformer

본 연구의 encoder와 decoder에 참조한 Transformer(Vaswani et al., 2017)는 그림 2처럼 간략한 구성도를 가진다.

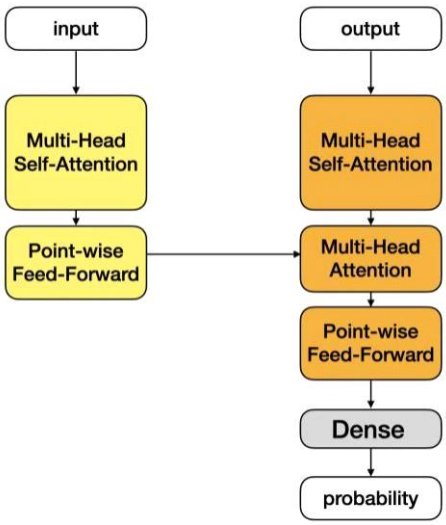

그림 2. 인코더와 디코더의 Transformer
Figure 2. Transformer

그림 2의 노란색으로 표시된 부분은 Transformer(Vaswani et al., 2017)의 encoder, 주황색으로 표시된 부분은 decoder이다. Transformer에서 주목할 점은 순차 데이터인 문자열을 다루는 모델임에도 불구하고 순환신경망을 사용하지 않는다는 점이다. 문자열을 처리하는 encoder와 decoder는 보통 input의 한 time step마다 특징을 추출하고 output을 한 time step 단위로 출력한다. Transformer는 autoregressive한 인공신경망을 사용하지 않기 때문에 input의 모든 time step을 통째로 처리하고 output의 모든 time step을 통째로 출력한다.

그러나 반대로 말해, Transformer는 시계열 데이터를 처리하는 순환신경망이나 이미지 데이터를 처리하는 합성곱신경망을 사용하지 않기 때문에 데이터를 처리할 때 데이터의 순서에 대한 파악이 이루어지지 않는다(Vaswani et al., 2017). 그러므로 시계열 데이터에 positional encoding을 합해서 전달한다. Positional encoding은 sine 함수와 cosine 함수를 이용해 데이터의 각 time step에 고유의 순서 정보를 삽입한다.

Transformer는 순환신경망과 합성곱신경망 대신 multi-head self-attention과 심층신경망으로 조직되었다. Attention은 두 데이터의 연관성을 학습하는 데 이용된다. 'I ate dinner.'와 '나는 저녁을 먹었다.'라는 문장을 가정하자. Attention은 'I'와 '나', 'ate'과 '먹었다', 'dinner'와 '저녁'이 각각 연관성이 높다는 정보를 학습시킨다. Self-attention은 동일한 데이터 내부의 연관성을 학습시킨다. 순차 데이터의 한 time step은 다른 time step의 영향을 받는다. Self-attention은 순차 데이터 내의 한 time step과 다른 time step과의 연관성을 토대로 데이터를 더욱 잘 나타낼 수 있는 특징을 출력한다. Multi-head attention은 데이터를 k개의 부분으로 나눈 후 각각 self-attention을 수행하고 다시 모아주는 역할을 한다. 환언하자면, 다층적인 attention으로 데이터와 데이터 간의 연관성을 더욱 다차원적으로 학습한다(Vaswani et al., 2017).

2.2. Length regulator

본 연구는 FastSpeech의 length regulator로 encoder와 decoder가 처리하는 특징 간의 연관성을 학습했다. Length regulator는 encoder가 출력한 특징의 길이를 decoder가 출력할 output 특징의 길이로 맞춘다. 본 연구의 모델은 문자열을 음성으로 출력하는 non-autoregressive 모델이다. 즉, encoder는 문자열을 통째로 처리하고, decoder는 문자열의 특징을 음성으로 통째로 만들어야 한다. 그러나 문자열과 음성의 길이가 다르기 때문에 모델 내부 연산 과정에 문제가 생긴다(Ren et al., 2019). 그러므로 encoder가 출력한 특징의 길이를 output으로 출력할 음성의 길이로 맞춰야 한다. 결국 모델은 길이 M의 문자열 input이 들어왔을 때 문자열과 대응하는 음성의 길이 N을 알고 있어야 한다. 예를 들면, '나는 밥을 먹었다.'라는 문자열을 소리 내어 읽는다면 몇 초의 음성이 되는지를 알아야 한다.

Length regulator는 학습이 잘 된 base TTS의 attention으로 음성의 길이를 학습한다. 본 연구에서는 Tacotron 2를 base TTS로 사용하지만 다른 모델이더라도 attention이 잘 나온다면 사용 가능하다. TTS는 attention으로 input 문자열과 output 음성 특징의 연관성을 학습시킨다. 예를 들면, '안녕.'이라는 문자열의 각 자모와 음성 특징의 각 프레임(frame)의 연결 관계를 학습시킨다. 이와 같이 훈련이 된 TTS의 attention은 각 문자열의 자모가 음성 특징의 몇 번째 프레임과 일치하는가에 연관된 정보를 담고 있다. Length regulator는 이 정보를 근거로 문자열 특징을 음성 특징의 길이만큼 늘린다. '안녕'이라는 문자열은 6개의 자모 'ㅇ', 'ㅏ', 'ㄴ', 'ㄴ', 'ㅕ', 'ㅇ'으로 표현된다. ㅇ이라는 문자열의 원소가 음성 특징의 1번부터 10번 frame과 연관된다면, 'ㅇ' 정보를 담은 encoder의 특징이 10번 반복된다. 다른 특징도 이와 같이 반복을 하면 음성 특징만큼의 길이로 늘어난다. Length regulator는 훈련 시 base TTS의 attention을 참고하지만 훈련이 끝나면 음성의 길이를 직접 예측해야 하기 때문에 심층신경망에 encoder가 출력한 특징을 넣어서 문자열의 각 time step과 대응하는 음성의 길이를 출력한다.

2.3. Super-resolution network

본 연구는 decoder의 output으로 mel-scale spectrogram을 출력시켰다. 현존하는 모델 중에는 mel-scale spectrogram을 음성으로 출력하는 인공신경망 모델이 많다. 그러나 대부분의 모델은 훈련 및 음성 출력 시간이 길다.

인공신경망 모델을 사용하지 않고 음성을 출력하는 대중적인 방법은 Griffin-Lim algorithm(Griffin & Lim, 1983)이다. Griffin-

Lim algorithm은 linear-scale spectrogram의 위상 정보를 추정해서 음성을 출력한다. 본 연구의 모델은 Griffin-Lim algorithm에 들어갈 linear-scale spectrogram을 출력하고자 DCTTS(Tachibana et al., 2017)의 super-resolution network를 사용한다. Super-resolution network는 decoder가 출력한 mel-scale spectrogram을 linear-scale spectrogram로 출력한다. Super-resolution network는 합성곱신경망과 transposed 합성곱신경망, highway network로 조직된다.

본 연구에서 이와 같이 linear-scale spectrogram을 바로 예측하는 대신, decoder로 mel-scale spectrogram을 예측한 후에 super-resolution network로 linear-scale spectrogram을 예측하는 이유는 아래와 같다. TTS는 저차원 문자열 정보를 고차원 음성의 형태로 압축을 푸는(decompress) 고난도의 작업을 수행한다(Wang et al., 2017). 문자열로부터 linear-scale spectrogram을 예측하는 과정이 바로 이와 같다. 반면, mel-scale spectrogram은 linear-scale spectrogram보다 정보량이 적기 때문에 문자열로부터 mel-scale spectrogram을 예측한 후 이로부터 linear-scale spectrogram을 예측하는 작업의 난이도는 상대적으로 낮아진다.

본 연구에서는 Tachibana et al.(2017)의 제안대로 mel-scale spectrogram을 실제 mel-scale spectrogram의 1/4 길이만큼 출력한다. 이 방법으로 모델의 연산량을 줄이고 출력 속도를 높일 수 있다. 모델은 완전한 mel-scale spectrogram의 맨 처음 time step부터 4칸 간격에 있는 time step들만 출력하도록 학습된다. Super-resolution network는 1/4 길이 mel-scale spectrogram을 완전한 linear-scale spectrogram로 출력하기 때문에 데이터의 길이를 역으로 증대하는 transposed 합성곱 신경망을 사용한다.

Super-resolution network가 출력한 linear-scale spectrogram은 Griffin-Lim algorithm의 input으로 들어간다. Griffin-Lim은 32번의 iteration을 돌고 나서 output으로 음성을 출력한다.

2.4. Training & inference

다음은 본 연구가 사용하는 모델의 학습 과정이다. Encoder는 문자열 input을 처리한다. Length regulator는 target mel-scale spectrogram에 대한 base TTS의 attention을 참고해서 encoder가 출력한 특징의 길이를 증대한다. 그리고 이와는 별개로 음성의 길이를 따로 예측한다. Decoder는 길이가 늘어난 encoder의 특징으로부터 mel-scale spectrogram을 출력한다. Super-resolution network는 decoder가 출력한 mel-scale spectrogram으로부터 linear-scale spectrogram을 출력한다. 모델은 decoder가 출력한 mel-scale spectrogram($\widehat{ms}$)과 target mel-spectrogram(ms)의 Mean Square Error, super-resolution network가 출력한 linear-scale spectrogram ($\widehat{ls}$)과 target linear-scale spectrogram(ls)의 Mean Square Error, length regulator가 출력한 음성의 길이($\widehat{dr}$)와 target 음성의 길이 (dr)의 Mean Square Error(식 1)로 parameter를 업데이트한다.

$$\frac{1}{k}\sum_{i=1}^{k}(ms_i-\widehat{ms}_i)^2+\frac{1}{k}\sum_{i=1}^{k}(ls_i-\widehat{ls}_i)^2+\frac{1}{k}\sum_{i=1}^{k}(dr_i-\widehat{dr}_i)^2 \quad (1)$$

Target mel-scale spectrogram과 linear-scale spectrogram은 target 음성에서 50ms frame length와 10ms frame hop length로 뽑았다.

모델이 학습 완료 후 음성을 생성하는 과정은 학습 과정과 크게 다르지 않다. Encoder는 문자열 input을 계산해서 length regulator에 전달한다. Length regulator는 base TTS의 보조 없이 output 음성의 길이를 예측해서 encoder 특징을 늘린다. Decoder는 length regulator 특징을 처리해서 super-resolution network에 보낸다. Super-resolution network가 출력한 linear-scale spectrogram은 Griffin-Lim algorithm에 의해 음성으로 출력된다.

3. 평가

3.1. 모델

본 연구의 모델과 autoregressive TTS의 Mean Opinion Score (MOS), 출력 속도, 안정성을 비교하는 실험으로 모델을 평가했다. 비교되는 모델은 Tacotron 2 – WaveNet으로 만들어진 autoregressive TTS이다. 최대한 공정한 환경을 조성하고자 본 연구의 모델과 비교되는 모델은 모두 batch 16으로 60만 step을 돌았다. 본 연구의 모델과 Tacotron 2의 optimizer parameter는 Shen et al.(2017)을 따랐다. WaveNet의 optimizer parameter는 van den Oord et al.(2016)을 따랐다. 본 연구의 모델은 비교되는 모델과 동일한 환경의 Tacotron 2의 attention을 길이 예측에 참조했다.

모든 모델은 자체 수집한 한국 성인 남성의 sample rate 22,050 Hz 11시간 음성 데이터로 학습했다. 본 연구의 모델과 비교되는 모델의 input은 음소열로 치환하지 않은 문자열이다.

3.2. Mean opinion score

TTS의 성능은 음성의 자연스러움(naturalness), 즉 인간의 음성과의 유사성과 이해도(intelligibility), 즉 단어를 알아듣기 쉬운 정도로 측정된다 (Dvorak, 2011). 두 기준은 객관적으로 측정하기 어렵기 때문에 MOS라는 인간의 주관적인 점수의 평균으로 평가된다(Holmes & Holmes, 2002). 본 연구는 자연스러움의 정도를 5점으로 나누었다. 1점(매우 부자연스러움), 2점(부자연스러움), 3점(보통), 4점(자연스러움), 5점(매우 자연스러움)이다. 대학생 및 대학원생 10명은 모델 당 15개의 음성과 ground truth 음성 15개에 점수를 부여했다. 표 1은 두 모델과 ground truth의 MOS 결과이다.

표 1. TTS와 실제 음성의 MOS
Table 1. MOS of synthesized speech & ground truth speech

Method	MOS (신뢰구간 95%)
Tacotron 2 - WaveNet	3.86±0.36
Research model	3.8±0.12
Ground truth	4.6±0.07

MOS, Mean Opinion Score; TTS, Text-to-Speech.

표 1에 의하면 본 연구의 모델과 Tacotron 2 - WaveNet의 MOS는 ground truth의 MOS보다 낮지만 4점(자연스러움)에 가깝다. 본 연구의 모델은 Tacotron 2 – WaveNet보다 MOS가 조금 낮지

만, 거의 비슷한 수준의 음성을 만들어낸다는 해석이 가능하다.

3.3. Inference 속도

표 2는 동일 환경에서 본 연구의 모델과 Tacotron 2 - WaveNet에게 200개의 문장을 출력시켰을 때의 각 모델의 속도를 ground truth와 비교한 비율이다. 각 모델은 NVIDIA의 A100-SXM4-40GB GPU 1개로 음성을 출력했다.

표 2. 실시간 음성과 TTS의 출력 속도 비율
Table 2. Inference speed rate of TTS compared with realtime speech

Method	Speed rate
Tacotron 2 - WaveNet	0.006x
Research model	600x
Ground truth	1x

TTS, Text-to-Speech.

표 2는 다음처럼 해석된다. 1분의 음성이 있을 때, 이 음성을 재생하면 1분이 걸린다. Tacotron 2 - WaveNet이 1분 음성을 출력하는 작업은 1분 음성을 재생하는 속도의 0.006x, 즉 약 3시간 정도 걸린다. 반면 본 연구의 모델이 1분 음성을 출력하는 작업은 실제의 600x, 즉 0.1초가 걸린다. Tacotron 2 - WaveNet은 full mel-scale spectrogram을 autoregressive하게 처리하지만, 본 연구의 모델은 1/4 길이 mel-scale spectrogram을 non-autoregressive하게 처리하는 차이에서 나온 결과로 판단된다.

3.4. 안정성

Ren et al.(2019)의 실험을 참조해서 두 모델의 안정성 실험을 진행했다. 동일 환경에서 본 연구의 모델과 Tacotron 2 – WaveNet에게 30개의 문장을 출력시켰다. 문장은 모델의 불안정성을 야기할 수 있는 것들로 추렸으며, Ren et al.(2019)을 따라서 1. 한 글자 문장(ex. 안.), 2. 흔하지 않은 자모 조합(ex. 뗣뛟껽.), 3. 반복되는 단어(ex. 각각각각각.), 4. 긴 문장의 네 종류이다. 표 3은 두 모델의 안정성 결과이다.

표 3. TTS의 오류율
Table 3. Error rate of TTS

Method	Error rate
Tacotron 2 - WaveNet	23%
Research model	8%

TTS, Text To Speech.

표 3의 본 연구의 모델의 오류율은 Tacotron 2 – WaveNet보다 낮다. 그러므로 불안정성을 야기하는 네 종류 문장에 있어 상대적으로 안정성이 높다고 판단된다. Tacotron 2 – WaveNet은 특히 반복되는 문장이 들어왔을 때 특정 time step부터 발음이 이상해지거나 긴 문장이 들어왔을 때 문장의 일부를 발음하지 않는 현상을 자주 보인다. Tacotron 2 – WaveNet에서 특정 time step이 잘못 예측될 때 이를 참고하는 뒤의 time step이 영향을 받기 때문인 것으로 해석된다.

4. 결론

본 연구는 autoregressive TTS의 단점을 보완하는 end-to-end non-autoregressive TTS를 제안했다. 다음은 본 연구의 모델이 autoregressive TTS와 비교해서 가지는 장점이다.

1. 안정성 실험 결과에 의하면, 본 연구의 모델은 음성을 통째로 출력하기 때문에 특정 time step이 잘못 출력될 때 그 뒤의 모든 time step이 잘못 출력되는 autoregressive TTS(Tacotron 2 - WaveNet)보다 안정적이다.
2. MOS 측정 실험 결과에 의하면, 본 연구의 모델은 autoregressive TTS(Tacotron 2 – WaveNet)와 비슷하게 자연스러운 음성을 생성한다.
3. 음성 출력 속도 실험 결과에 의하면, 본 연구의 모델은 autoregressive TTS(Tacotron 2 - WaveNet)보다 속도가 훨씬 빠르다.

본 연구의 모델의 성능, 안정성 및 속도는 Tacotron 2 – WaveNet과 비교되었으나, 현존하는 여러 autoregressive TTS와의 비교 실험이 필요하다. 특히, Tacotron과 같이 모델 1개로 제작된 autoregressive TTS와의 비교 실험도 가능하다. 추가로 FastSpeech - MelGAN 등 모델 2개로 제작된 non-autoregressive TTS와의 비교 실험, 모델 구조 또는 loss에 대한 ablation 실험을 진행할 수 있다.

본 연구의 모델을 개선하려면 다음의 연구가 진행되어야 할 것이다. 우선적으로는 모델의 성능을 높이는 연구가 있다. MOS 측정 실험의 결과에 의하면 본 연구의 모델은 아직 autoregressive TTS(Tacotron 2 – Wavenet)의 성능을 완전히는 따라잡지 못한다. 현재의 속도를 유지하면서 성능을 올릴 수 있는 방법을 찾아야 한다.

다음으로는 모델의 속도를 높이는 연구가 가능하다. 본 연구의 모델은 문자열에서 mel-scale spectrogram을 거쳐서 linear-scale spectrogram을 출력한다. Linear-scale spectrogram의 정보량 문제 때문에 문자열을 바로 linear-scale spectrogram으로 출력하는 모델이 현재 많이 나와 있지 않을뿐더러 성능이 높다고 알려진 모델이 없다. 본 연구의 모델이 mel-scale spectrogram을 예측하는 부분 없이도 문자열로부터 바로 linear-scale spectrogram을 제대로 출력할 수 있다면, 훈련 parameter가 감소해서 모델의 속도가 더 빨라질 것이다.

마지막으로 데이터에 관련된 모델의 안정성을 높이는 연구가 필요하다. 데이터 A, B, C를 가정했을 때 모델이 데이터 A로는 학습을 잘 할 수 있지만 데이터 B, C로는 학습을 못할 수도 있다. 각 데이터 별로 특성이 다르기 때문이다. 현재의 모델은 성인 남성 데이터 1개로 학습이 잘 된다는 점이 확인되었지만 추가 연구로 더 많은 데이터에 대한 안정성의 확보를 확인할 수 있다.

References

Arik, S. O., Chrzanowski, M., Coates, A., Diamos, G., Gibiansky, A.,

Kang, Y., Li, X., ... Shoeybi, M. (2017). Deep voice: Real-time neural text-to-speech. Retrieved from https://arxiv.org/abs/1702.07825

Cho, K. (2013). Boltzmann machines and denoising autoencoders for image denoising. Retrieved from https://arxiv.org/abs/1301.3468

Dvorak, J. L. (2011). *Moving wearables into the mainstream: Taming the Borg.* New York, NY: Springer.

Griffin, D., & Lim, J. (1983, April). Signal estimation from modified short-time Fourier transform. *Proceedings of the 8th International Conference on Acoustics, Speech, and Signal Processing* (pp. 804-807). Boston, MA.

Holmes, J., & Holmes, W. (2002). *Speech synthesis and recognition*. London, UK: CRC Press.

Kumar, K., Kumar, R., de Boissiere, T. Gestin, L., Teoh, W. Z., Sotelo, J., de Brebisson, A., ... Courville, A. (2019). MelGAN: Generative adversarial networks for conditional waveform synthesis. Retrieved from https://arxiv.org/abs/1910.06711

Ren, Y., Ruan, Y., Tan, X., Qin, T., Zhao, S., Zhao, Z., & Liu, T. Y. (2019). FastSpeech: Fast, robust and controllable text to speech. Retrieved from https://arxiv.org/abs/1905.09263

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., ... Wu, Y. (2017). Natural TTS synthesis by conditioning Wavenet on mel spectrogram predictions. Retrieved from https://arxiv.org/abs/1712.05884

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. Retrieved from https://arxiv.org/abs/1409.3215

Tachibana, H., Uenoyama, K., & Aihara, S. (2017). Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention. Retrieved from https://arxiv.org/abs/1710.08969

Valle, R., Shih, K., Prenger, R., & Catanzaro, B. (2020). Flowtron: An autoregressive flow-based generative network for text-to-speech synthesis. Retrieved from https://arxiv.org/abs/2005.05957

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. Retrieved from https://arxiv.org/abs/1706.03762

van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., ... Kavukcuoglu, K. (2016). WaveNet: A generative model for raw audio. Retrieved from https://arxiv.org/abs/1609.03499

van den Oord, A., Li, Y., Babuschkin, I., Simonyan, K., Vinyals, O., Kavukcuoglu, K., van den Driessche, G., ... Hassabis, D. (2017). Parallel WaveNet: Fast high-fidelity speech synthesis. Retrieved from https://arxiv.org/abs/1711.10433

Wang, T., Liu, X., Tao, J., Yi, J., Fu, R., & Wen, Z. (2020, October). Non-autoregressive end-to-end TTS with coarse-to-fine decoding. *Proceedings of the 21st Annual Conference of the International Speech Communication Association* (pp. 3984-3988). Shanghai, China.

Wang, Y., Skerry-Ryan, RJ., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., ... Saurous, R. A. (2017). Tacotron: Towards end-to-end speech synthesis. Retrieved from https://arxiv.org/abs/1703.10135

Yamamoto, R., Song, E., & Kim, J. M. (2019). Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. Retrieved from https://arxiv.org/abs/1910.11480

Yarrington, D. (2007). Synthesizing speech for communication devices. In K. Greenebaum, & R. Barzel (Eds.), *Audio anecdotes: Tools, tips and techniques for digital audio* (Vol. 3, pp. 143-155). Wellesley, MA: AK Peters.

• **김위백 (Wiback Kim)**
고려대학교 문과대학 영어영문학과 박사과정
서울시 성북구 안암로 145
Tel: 02-3290-1980
Email: kwb425@korea.ac.kr
관심분야: 음성학, 언어공학

• **남호성 (Hosung Nam)** 교신저자
고려대학교 문과대학 영어영문학과 교수
서울시 성북구 안암로 145
Tel: 02-3290-1991
Email: hnam@korea.ac.kr
관심분야: 음성학, 음운론, 언어과학, 언어공학

End-to-end 비자기회귀식 가속 음성합성기

김 위 백 · 남 호 성

고려대학교 영어영문학과

국문초록

Autoregressive한 TTS 모델은 불안정성과 속도 저하라는 본질적인 문제를 안고 있다. 모델이 time step t의 데이터를 잘못 예측했을 때, 그 뒤의 데이터도 모두 잘못 예측하는 것이 불안정성 문제이다. 음성 출력 속도 저하 문제는 모델이 time step t의 데이터를 예측하려면 time step 1부터 t-1까지의 예측이 선행해야 한다는 조건에서 발생한다. 본 연구는 autoregression이 야기하는 문제의 대안으로 end-to-end non-autoregressive 가속 TTS 모델을 제안한다. 본 연구의 모델은 Tacotron 2 – WaveNet 모델과 근사한 MOS, 더 높은 안정성 및 출력 속도를 보였다. 본 연구는 제안한 모델을 토대로 non-autoregressive한 TTS 모델 개선에 시사점을 제공하고자 한다.

핵심어: 딥러닝, 인공신경망, 음성합성, Text-to-Speech (TTS)