pISSN 2005-8063
eISSN 2586-5854
2018. 03. 31.
Vol.10 No.1
pp. 39-48

말소리와 음성과학

Phonetics and Speech Sciences

한국음성학회지

https://doi.org/10.13064/KSSS.2018.10.1.039

한국어 text-to-speech(TTS) 시스템을 위한 엔드투엔드 합성 방식 연구*

An end-to-end synthesis method for Korean text-to-speech systems

최 연 주 · 정 영 문 · 김 영 관 · 서 영 주 · 김 회 린**

Choi, Yeunju · Jung, Youngmoon · Kim, Younggwan · Suh, Youngjoo · Kim, Hoirin

Abstract

A typical statistical parametric speech synthesis (text-to-speech, TTS) system consists of separate modules, such as a text analysis module, an acoustic modeling module, and a speech synthesis module. This causes two problems: 1) expert knowledge of each module is required, and 2) errors generated in each module accumulate passing through each module. An end-to-end TTS system could avoid such problems by synthesizing voice signals directly from an input string. In this study, we implemented an end-to-end Korean TTS system using Google's Tacotron, which is an end-to-end TTS system based on a sequence-to-sequence model with attention mechanism. We used 4392 utterances spoken by a Korean female speaker, an amount that corresponds to 37% of the dataset Google used for training Tacotron. Our system obtained mean opinion score (MOS) 2.98 and degradation mean opinion score (DMOS) 3.25. We will discuss the factors which affected training of the system. Experiments demonstrate that the post-processing network needs to be designed considering output language and input characters and that according to the amount of training data, the maximum value of *n* for *n*-grams modeled by the encoder should be small enough.

Keywords: attention mechanism, end-to-end, Korean text-to-speech system, sequence-to-sequence, Tacotron

1. 서론

Text-to-speech(TTS) 시스템이란 텍스트가 입력되어서 그에 대응하는 음성으로 변환되어 출력되는 시스템으로, 음성 합성 시스템이라고도 불린다. 여기서 중요한 점은 출력되는 합성음이 실제 사람이 말하는 것처럼 충분히 자연스러워야 한다는 점이다. 사람은 어떤 생각을 언어로 변환한 뒤 조음 기관에서 발성하는 음성 신호로서 생각을 내뱉는다. 이를 표방하는 TTS 시스템은 반드시 텍스트 분석부와 음성 합성부를 가지게 된다.

초기의 음성 합성 시스템은 1세대로서 포만트 합성기, 2세대로서 선형 예측 부호화(LPC, linear predictive coding) 기반의 음성 합성기 등의 규칙 기반의 방식이 주로 사용되었다. 2세대는 1세대에 비해 명료도는 좋아졌으나, 자연성에 여전히 한계가 있었다(Rabiner & Schafer, 2011). 규칙 기반의 방식으로 더 이상 발전

* 이 논문은 산업통상자원부의 산업기술혁신사업으로부터 지원을 받아 수행된 연구입니다(지원번호: 10080667, 음원 다양화를 통하여 로봇의 감정 및 개성을 표현할 수 있는 대화음성합성 원천기술 개발).

** 한국과학기술원, hoirkim@kaist.ac.kr, 교신저자

Received 5 February 2018; Revised 9 March 2018; Accepted 21 March 2018

의 여지가 보이지 않자, 컴퓨터의 연산 속도 및 메모리의 증가에 따라 3세대부터는 데이터 기반의 방식이 많이 연구되었다.

그 중 먼저 널리 연구되고 현재 상용화되어 있는 방식은 음편 선정(unit selection) 방식이다(Hunt & Black, 1996). 이 방식은 짧은 단위의 음편들을 저장한 후 텍스트에 해당하는 음편들을 선택해 연결하여 합성음을 출력한다. 실제 사람의 음성을 녹음한 음편을 사용하기 때문에 음질이 좋다는 장점이 있지만, 많은 양의 데이터를 사용함에도 불구하고, 연결한 두 음편 사이의 경계가 부자연스럽다는 문제, 주어진 문장에 대해 항상 똑같은 발화만이 가능하다는 문제 등이 존재한다.

이러한 한계점들을 극복하고자 통계적 파라미터 방식 음성 합성(statistical parametric speech synthesis) 시스템이 제안되었다. 대표적인 예로 은닉 마르코프 모델(HMM, hidden Markov model) 기반 TTS 시스템(HTS, HMM-based speech synthesis)이 있다. 음편 선정 방식과는 대조적으로 적은 양의 데이터만으로도 TTS가 가능하며, 파라미터를 조절해서 감정이 들어간 음성을 합성하거나, 화자의 목소리를 변환하는 등의 다양한 음성 합성이 가능하다(Tokuda *et al*., 2013).

최근 10여 년 사이에 심층 신경망(DNN, deep neural network)을 활용해 성능을 크게 향상시킨 연구 결과가 기계 번역, 음성 인식 등의 다양한 분야에서 나타나면서 음성 합성에서도 DNN 기반의 연구 결과들이 발표되고 있다.

Merlin은 영국 에든버러 대학의 CSTR(The Centre for Speech Technology Research)에서 개발한 오픈 툴킷으로, 파라미터를 이용한 통계적 음성 합성을 위한 DNN을 구성하는 것이 그 목적이다(Wu *et al*., 2016). 이 시스템은 음향 모델링에 해당하는 부분만을 구현했기 때문에 이전 단계의 텍스트 분석부와 이후 단계의 음성 합성부를 조합하여 사용해야 한다.

구글 딥마인드(DeepMind)의 Oord *et al*.(2016)은 주로 사용되어왔던 신호처리 기반의 음성 합성부를 DNN 기반의 WaveNet이라는 모델로 새롭게 구성하였다. 이 모델은 음성의 샘플 단위로 연산을 수행하며, dilated causal convolution이라는 새로운 방법을 제시했다. 이로써 텍스트 분석부를 통해 출력한 언어 특징을 WaveNet에 입력으로 넣어주면, 별도의 음향 모델링 없이도 음성 신호를 합성할 수 있다. WaveNet이 합성한 음성의 음질은 독보적이지만, 훈련 및 합성 속도가 느리다는 단점이 있다.

바이두(Baidu)는 Deep Voice 이후 Deep Voice 2, Deep Voice 3까지 계속해서 Deep Voice 시리즈를 발표하고 있다(Arik *et al*., 2017a, 2017b; Ping *et al*., 2017). Deep Voice는 기존의 통계적 음성 합성 시스템의 모든 부분을 DNN 기반 방식으로 구현한 TTS 시스템이고, Deep Voice 2는 Deep Voice의 각 부분들을 발전시키면서 화자의 정보를 나타내는 벡터를 활용하여 구현한 다중 화자 TTS 시스템이다. Deep Voice 3는 Deep Voice 2에서 다중 화자 음성 합성의 원리만 그대로 이용하면서, 전체 구조를 CNN (convolutional neural network)으로 구성한 attention 메커니즘 기반의 TTS 시스템으로, 저자는 attention 메커니즘에서의 오류를 줄이기 위한 방법과 음성 합성부를 Griffin-Lim 방식, WORLD, WaveNet 방식으로 구성했을 때의 음질의 차이 등 다양한 방면에서의 연구를 진행했다(Bahdanau *et al*., 2014; Griffin & Lim, 1984; Morise *et al*., 2016).

이러한 음성 합성 시스템 방식들은 모두 하나의 시스템 안에서 여러 개의 모듈을 사용한다는 공통점이 있다. 이는 각 모듈에 대한 전문적인 지식을 요구하기 때문에 진입장벽이 높다는 문제점과 각 모듈에서의 loss가 누적될 수밖에 없다는 문제점을 야기한다. 반면 엔드투엔드(end-to-end) 시스템은 입력부터 출력까지 하나의 모듈로 이루어진 시스템이다. 따라서 기존의 방식과는 달리 각 모듈에 대한 전문적인 지식이 필요하지 않아 진입장벽이 낮고, 각 모듈에서의 loss가 누적되는 문제가 해결된다.

이에 따라 2017년 3월, 구글이 Tacotron을 발표했다(Wang *et al*., 2017). Tacotron은 입력 문자열에서 스펙트로그램을 출력하는 attention 메커니즘 기반의 순환신경망(RNN, recurrent neural network) 인코더-디코더와 음성 합성부로 이루어져 있다(Cho *et al*., 2014; Sutskever *et al*., 2014). Tacotron에서 사용된 RNN 인코더-디코더 모델은 주로 sequence-to-sequence 모델이라고 불리며,

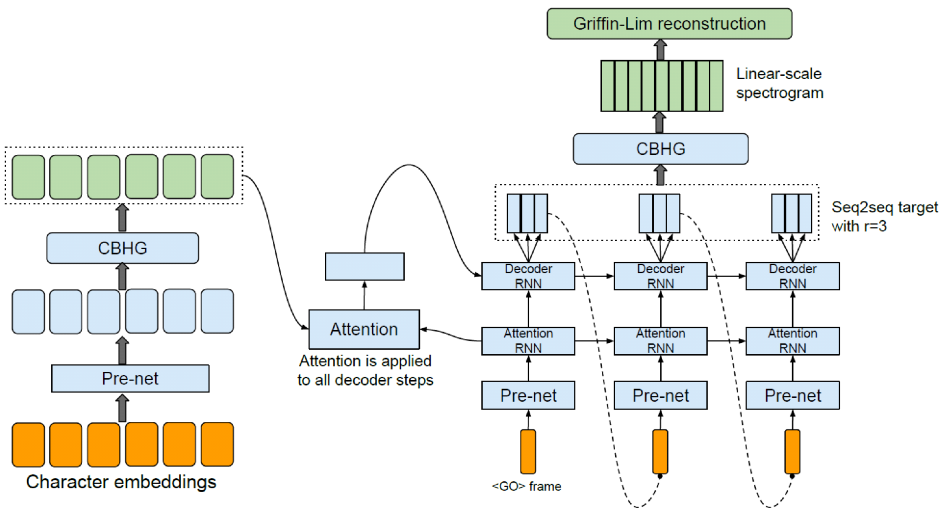

그림 1. Tacotron의 전체 구조
Figure 1. Structure of Tacotron

기계 번역 분야에서 처음 제안된 이후에 다양한 분야에서 뛰어난 성능을 보이고 있다. Tacotron은 사전 훈련을 필요로 하지 않기 때문에 현존하는 TTS 시스템 중에 end-to-end 시스템으로서의 특징을 가장 잘 나타낸다. 이어서 12월에는 Shen *et al.*(2017)이 Tacotron 2를 발표했으며, Tacotron의 문제점이었던 attention 메커니즘과 음성 합성 알고리즘을 개선하여 현존하는 최고 음질의 합성음을 출력하였다.

본 연구는 엔드투엔드 합성 방식을 한국어 TTS에 적용하는 방법과, 그 적용 결과를 분석하여 제시한다. 전반적으로 Wang *et al.*(2017)이 제안한 방법론을 기반으로 하나, 보다 적은 데이터를 사용하여 자연스러운 한국어 합성음을 생성하기 위한 시스템을 구현하는 것을 목표로 한다.

2. 엔드투엔드 한국어 TTS 시스템

2.1. 입출력

End-to-end 한국어 TTS 시스템의 입출력은 다음과 같다. 입력은 문자 임베딩 열이며, 훈련 및 합성 과정에서 사용한 문자는 초성 19개, 중성 21개, 종성 27개와 문장부호 13개로 총 80개이다. 입력 문장에 대한 텍스트 정규화는 다음과 같다. (1) 영어 단어를 포함하여 '119 구급차(일일구 구급차)'나 '1+1(원플러스원)'과 같이 일반적인 경우와 다르게 발성하는 단어, 그 외 '10-15분'과 같이 발성할 수 있는 방법이 여러 가지 있는 단어 등은 미리 사전에 정의해놓고, 입력 문장 중 사전에 있는 단어는 사전에 표기된 대로 바꾼다. (2) 아라비안 숫자를 한글로 바꾼다. (3) jamo라는 파이썬 패키지를 사용해 한글 초, 중, 종성 열로 변환한다.

디코더의 출력은 효율적인 학습을 위해 80 밴드의 멜 스케일 스펙트로그램으로 사용한다. 디코더의 표적으로 바로 선형 스케일 스펙트로그램과 같은 고차원의 표적을 설정할 경우, 연산량이 많아지고, 필요 이상의 정보가 많아서 정밀한 디코딩이 어렵기 때문이다. 이후에 음성 신호를 합성하기 위해 1025차 선형 스케일 스펙트로그램으로 변환한 뒤 최종적으로 음성 신호를 출력한다.

2.2. 구조

기본적으로 <그림 1>에 해당하는 Wang *et al.*(2017)이 제안한 Tacotron의 구조를 따른다. Tacotron은 attention 메커니즘 기반의 RNN 인코더-디코더 구조를 중심으로 디코더의 출력인 80 밴드의 멜 스케일 스펙트로그램을 1025차 선형 스케일 스펙트로그램으로 변환하는 후처리 네트워크와 그로부터 합성음을 출력하는 음성 합성 알고리즘까지 포함하고 있다. 이때 후처리 네트워크는 Wang *et al.*(2017)과 달리 highway 네트워크를 사용한다(Srivastava *et al.*, 2015).

2.2.1. CBHG 모듈

<그림 2>에 해당하는 CBHG 모듈은 Lee *et al.*(2016)이 제안한 기계 번역을 위한 인코더로부터 착안된 구조로, 1차 convolution bank, highway 네트워크, bidirectional gated recurrent unit(GRU)로

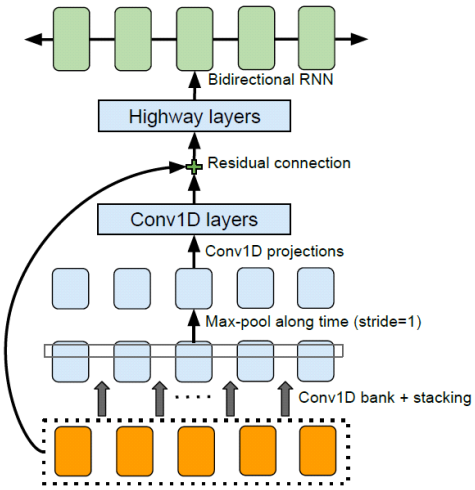

그림 2. CBHG 구조
Figure 2. Structure of CBHG
CBHG, 1-D convolution bank + highway network + bidirectional gated recurrent unit

이루어져 있고, 이를 줄여서 CBHG라고 부른다(Srivastava *et al.*, 2015; Cho *et al.*, 2014; Chung *et al.*, 2014). 1차 convolution bank에서는 unigram부터 K-gram까지를 모델링하기 위해 1부터 K까지의 길이를 가지는 필터로 입력을 convolution하고, 그 결과들을 쌓는다. 그리고 local invariance를 키우기 위해 max pooling을 한다. 여기서 local invariance를 키운다는 것은, 문맥이 달라져도 변하지 않는 부분들을 강조한다는 것으로 볼 수 있다. 이때 시간축 상의 해상도(resolution)를 유지하기 위해 stride=1로 한다. 이후 high-level feature들을 뽑기 위해 projection이라고 부르는 몇 층의 1차 convolution을 거친 뒤 highway 네트워크까지 거치도록 한다. 이때 모든 1차 convolution은 batch normalization을 함께 사용하여 internal covariate shift 문제를 해결한다(Ioffe & Szegedy, 2015).

Projection 후에 residual connection을 적용하여 1차 convolution들의 결과에 처음 입력을 더한 값이 highway 네트워크의 입력으로 들어가게 된다(He *et al.*, 2016). Residual connection은 보통 상당히 깊은 구조에서 훈련의 수렴을 돕는다고 알려져 있는데, 여기에서는 깊지 않은 구조임에도 불구하고 수렴과 일반화에 도움이 되는 것을 확인했다. Highway 네트워크는 입력이 x, 출력이 y라고 할 때, 아래와 같이 한 층의 신경망을 거친 결과, H(x, W_H)와 원래 입력 두 값을 weighted sum하는 구조이다.

$$y = H(x, W_H) \cdot T(x, W_T) + x \cdot C(x, W_C) \quad (1)$$

이때 Tacotron은 C(x, WC) 대신 1-T(x, WT)를 사용하고, W_H와 W_T는 모두 훈련을 통해 구한다. 이 역시 모델의 일반화에 도움이 된다. Highway 네트워크의 결과가 최종적으로 bidirectional GRU의 입력이 된다. Bidirectional GRU의 forward annotation vector와 backward annotation vector를 연결한 벡터가 최종적으로 입력의 annotation vector가 된다.

2.2.2. 인코더

인코더는 입력 문자 임베딩 열을 받아 annotation vector를 출력하는 부분으로서, Tacotron은 보다 robust한 인코더를 구현하기 위해 RNN이 아닌 CBHG를 사용하고, 그 전에는 입력 문자 임베딩 열이 pre-net을 거치도록 한다. 이때 입력 문자 임베딩 열은 각 문자를 one-hot 벡터로 변환한 뒤, 연속 벡터로 변환한 결과를 나열한 것이다. Pre-net은 dropout 기법을 적용한 2층의 fully connected layer로서 과적합(overfitting)을 방지하고 훈련이 수렴하는 것을 돕는다(Srivastava *et al*., 2014).

2.2.3. 디코더

디코더는 특정 time step 프레임의 스펙트로그램을 입력으로 받고, 다음 time step 프레임의 스펙트로그램을 출력한다. 본 연구에서는 Bahdanau *et al*.(2014)과는 달리, Vinyals *et al*.(2015)이 제안한 논문에서와 같이 attention RNN을 따로 두는 방식의 디코더를 사용하며, 인코더에서와 마찬가지로 입력 스펙트로그램은 우선 pre-net을 거치고, 그 결과가 attention RNN의 입력이 된다. Attention RNN의 hidden state는 annotation vector와 함께 alignment 모델의 입력으로 들어가고, alignment 모델의 출력인 context vector와 attention RNN의 hidden state가 decoder RNN의 입력으로 들어가게 된다.

즉, 인코더의 hidden state를 $(h_1, ..., h_T)$, attention RNN의 hidden state를 $(s_1, ..., s_{T'})$라고 할 때 alignment 모델은 다음과 같이 계산한다.

$$e_{ij} = a(s_i, h_h) = v_a^T \tanh(W_a s_i + U_a h_j) \quad (2)$$

이에 따라 $\alpha_{ij} = softmax(e_{ij})$, $c_i = \sum_{j=1}^{T} \alpha_{ij} h_j$가 되고, c_i는 attention RNN의 hidden state와 concatenate되어 디코더 RNN의 조건부 입력으로 쓰인다.

첫 디코더 time step에서는 <GO> 프레임이라는 모든 값이 0인 스펙트로그램이 입력으로 쓰인다. Attention RNN으로는 256-unit GRU 1층을, decoder RNN으로는 residual connection을 포함한 256-unit GRU 2층을 사용한다. Residual connection은 모델이 더 빨리 수렴하기 위해 필요하다.

이때 중요한 설정은 디코더 time step 당 하나가 아닌 여러 프레임의 스펙트로그램을 예상함으로써 훈련 시간, 합성 시간, 모델 사이즈를 줄이는 것이다. 이는 연속한 프레임의 스펙트로그램끼리 서로 겹치는 정보가 많기 때문에 가능하다. 이렇게 디코더 time step 당 예측하는 프레임의 개수를 reduction factor(r)라고 부른다. 본 연구에서는 r이 4-10일 때 작동함을 확인했다.

2.2.4. 후처리 및 음성 합성

디코더의 출력이 멜 스케일이므로 이를 선형 스케일로 변환하기 위해 후처리 네트워크를 사용한다. 후처리 네트워크는 디코더의 출력을 모든 time step에 대해 고려할 수 있다는 장점을 가진다. 본 연구에서는 Wang *et al*.(2017)과 달리 후처리 네트워크로의 스케일 변환이라는 간단한 목적과 훈련 데이터양의 제약에 따라, CBHG 모듈에서 convolution bank와 bidirectional GRU를 제외한 2층의 256-unit highway 네트워크를 사용한다.

선형 스케일 스펙트로그램을 음성 신호로 합성하는 데에는 Griffin-Lim 알고리즘을 사용한다. 이 알고리즘은 다음과 같이 반복적인 과정을 통해 주어진 modified STFT magnitude (MSTFTM)와 가장 비슷한 STFT magnitude(STFTM)을 가진 음성 신호를 복원하는 알고리즘이다.

1. 이전 단계에서 출력된 음성 신호의 STFT를 계산한 뒤 진폭을 입력으로 주어진 MSTFTM으로 대체한다.
2. 새로운 STFT의 진폭과 입력 MSTFT의 진폭의 squared error가 최소가 되도록 원래 신호를 복원한다.
3. 1과 2를 반복한다.

일반적인 보코더는 소스-필터 모델을 기반으로 구성되어 특유의 윙윙거리는 소리 혹은 쉰 소리가 합성음에 포함되고, 위상을 쓰지 않는 대신 F0와 duration 정보가 필요하다. 반면, 이 알고리즘은 특정 모델을 가정하지 않기 때문에 윙윙거리는 소리가 합성음에 포함되지 않고, F0와 duration 정보 없이 단순한 반복 과정을 통해 위상을 복원하여 음성 신호를 출력하기 때문에 계산량에 있어서는 일반적인 보코더보다 훨씬 유리하지만, 음성의 명료도가 떨어진다.

2.3. 훈련

Loss로서 디코더의 멜 스케일 스펙트로그램의 L1 loss와 후처리 네트워크의 선형 스케일 스펙트로그램의 L1 loss의 가중치 합을 사용한다. 이때 두 L1 loss의 가중치는 같고, 선형 스케일 스펙트로그램의 L1 loss는 3,000 Hz 이하의 값들에 대해 아래와 같이 가중치를 둬서 사용한다. 이는 두 L1 loss의 가중치가 다를 때, L1 loss 대신 L2 loss를 사용했을 때, 혹은 선형 스케일 스펙트로그램의 L1 loss에서 5,000 Hz 이하의 값들에 가중치를 둘 때보다 alignment가 더 잘 됨을 실험적으로 확인했다.

$$n\_priority\_freq = floor(3000 / (sampling\ rate * 0.5) * dimension\ of\ linear-scale\ spectrogram) \quad (3)$$

$$linear\ loss = 0.5 * average(l1) + 0.5 * average(l1[1 : n\_priority\_freq]) \quad (4)$$

β_1=0.9, β_2=0.99, ϵ=10^{-8}인 Adam optimizer를 사용해 최적화를 한다. 초기 학습률(learning rate)은 0.002로 아래와 같은 learning rate decay를 적용한다.

$$warmup\ steps = 2000 \quad (5)$$

$$learning\ rate = initial\ learning\ rate \cdot warmup\ steps^{0.5} \cdot \min(step \cdot warmup\ steps^{-1.5}, step^{-0.5}) \quad (6)$$

3. 실험

3.1. 실험 환경

본 연구에서는 단일 여성 화자 한국어 데이터베이스를 사용해 end-to-end 한국어 TTS 시스템을 구현한다. 데이터베이스는 잡음이 거의 없는 사무실 환경에서 전문 성우가 발화한 4,392개의 문장을 16 kHz sampling rate으로 녹음한 16bit 음원들로 이루어져 있다. Arik *et al.*(2017b)이 각 음원에서의 목소리 시작 타이밍이 다르면 Tacotron이 잘 훈련되지 않는다고 밝힘에 따라 앞뒤 묵음을 제거하여 총 9.45시간 분량이며, 이는 Wang *et al.*(2017)이 Tacotron을 훈련시킬 때 사용한 데이터베이스 양의 37%에 해당하는 적은 양이다. 이 중 훈련 데이터로는 총 9.04시간 분량의 4,000문장을 사용하고, 검증 데이터로 0.41시간 분량의 372문장을, 실험 데이터로는 나머지 20문장을 사용한다. Griffin-Lim 알고리즘의 반복 횟수는 100번으로 설정하고, 이를 통과해서 나온 합성음은 0.8초 이상의 침묵이 나타나면 그 이후가 모두 제거되어 최종 합성음으로 출력된다.

Baseline으로는 기존의 통계적 파라미터 방식 음성 합성 시스템과의 비교를 위해 가장 대표적인 예인 HTS 방식 음성 합성 시스템을 사용한다. 데이터베이스는 end-to-end 시스템과 동일한 것을 사용하나, 앞뒤 묵음과 단어 사이 묵음도 모델링하기 때문에 앞뒤 묵음을 제거하지 않고 사용한다. 또한 텍스트 정규화를 하지 않고, 미리 입력 텍스트에서 문장부호는 제거하고 아라비아 숫자는 한글로 변환하여 훈련한다. 보코더로는 STRAIGHT (Speech Transformation and Representation using Adaptive Interpolation and weiGHTed spectrum)를 사용한다(Kawahara, 1997). 이에 따라 음성 특징 벡터로 44차 mel-generalized cepstral coefficients(MGC)와 1차, 2차 미분값, 더불어 26차 BAP(band aperiodicity)와 F0를 파라미터로 사용한다. Tacotron을 기반으로 구현한 end-to-end 한국어 TTS 시스템의 자세한 하이퍼파라미터들은 <표 1>에 나타나 있다. 실험을 통해 최적의 값을 찾았으며, Arik *et al.*(2017b)이 밝혔듯이 모델이 하이퍼파라미터와 데이터에 예민하기 때문에 제시한 튜닝이 완벽하지 않을 수 있다. 흥미로운 점은 같은 모델을 LJ Speech 데이터셋을 이용하여 훈련할 때는 프레임 길이와 오버랩 길이가 각각 50 ms, 12.5 ms일 때와 100 ms, 25 ms일 때 모두 alignment 모델이 수렴했는데, 한국어 데이터에 대해서는 100 ms, 25 ms일 때만 alignment 모델이 수렴하였다. 요인으로는 언어, 화자, 데이터의 sampling rate 차이 등이 가능하지만, 비교 자료가 부족하여 결론을 내리기는 어렵다. 또한 reduction factor는 4일 때와 5일 때를 비교했을 때, 5일 때가 일반적으로는 청취 성능이 더 좋았으나, 문장에 따른 기복이 더 심하여 평가할 때는 4를 택하였다.

3.2. 주관적 음질 평가

표 2. 5-스케일 주관적 음질 평가 결과
Table 2. 5-scale subjective evaluation results

Model	MOS	DMOS
HTS	3.96±0.52	3.86±0.53
Tacotron	2.98±1.02	3.25±0.92

HTS, HMM-based speech synthesis; DMOS, degradation mean opinion score; MOS, mean opinion score

20에서 30대까지의 정상 청력을 가진 남녀 11명을 대상으로 20문장에 대한 합성음의 mean opinion score(MOS) 평가와 degradation

표 1. End-to-end TTS 시스템 하이퍼파리미터 설정
Table 1. Detailed hyper-parameters of the end-to-end text-to-speech system

Spectral 분석	Pre-emphasis: 0.97, 프레임 길이: 100 ms, 오버랩 길이: 25 ms, 윈도우 종류: Hann
사용한 문자 개수	80개
문자 임베딩	128차
인코더 CBHG	Conv1D bank: K=5, conv-k-64-ReLU Max pooling: stride=1, width=2 Conv1D projections: conv-3-128-ReLU → conv-3-128-linear Highway network: 2 layers of FC-128-ReLU Bidirectional GRU: 128 cells
인코더 pre-net	FC-128-ReLU → Dropout(0.5) → FC-128-ReLU → Dropout(0.5)
디코더 pre-net	FC-128-ReLU → Dropout(0.5) → FC-128-ReLU → Dropout(0.5)
디코더 RNN	2-layer residual GRU(256 cells)
Attention RNN	1-layer GRU(256 cells)
Reduction factor (r)	4
후처리 highway network	2-layers of FC-256-ReLU
전처리에서 제거한 침묵 기준	6 dB 이하
합성음에서 제거한 침묵 기준	-40 dB 이하

CBHG, 1-D convolution bank + highway network + bidirectional gated recurrent unit; Conv1D, 1-D convolution; FC, fully-connected; conv-k-c-ReLU, 1-D convolution with width k and c output channels with ReLU activation; (길이 k의 필터와 c개의 출력 채널을 가지고, ReLU(rectified linear unit)를 비선형 함수로서 사용하는 1차 convolution); GRU, gated recurrent unit; RNN, recurrent neural network; TTS, text-to-speech

mean opinion score(DMOS) 평가를 시행했다. MOS 평가는 합성음의 음질을 1-5점으로 절대 평가하는 평가이며, DMOS 평가는 합성음의 음질을 원본 음원과 비교하여 1-5점으로 상대 평가하는 평가이다. 주관적 음질 평가 결과는 <표 2>에 나와 있다. MOS와 DMOS 모두 HTS 방식 합성음이 Tacotron 기반 합성음보다 높으나, 다음 사항들을 고려해야 한다.

HTS 방식 TTS 시스템은 이미 오래 연구되어 와서 이제 완성형에 이르러 있으며, 애초에 적은 양의 훈련 데이터로도 훈련이 잘 되는 것이 목적인 시스템이다. 반면 end-to-end TTS 시스템은 이제 연구되기 시작했으며, 많은 양의 훈련 데이터를 필요로 하는데 보유한 데이터양의 한계로 적은 양의 데이터를 사용했다.

또한 DMOS와 MOS를 비교해보면 HTS 방식 합성음은 MOS에 비해 DMOS가 0.1점 낮은 반면, Tacotron 기반 합성음은 MOS에 비해 DMOS가 0.27점이 높았다. 이를 통해 억양은 Tacotron 기반 합성음이 HTS 방식 합성음보다 원래 화자의 억양과 더 비슷하다는 것을 알 수 있다. 즉, 자연성은 Tacotron 기반 합성음이 HTS 방식 합성음보다 높다. 적은 양의 훈련 데이터를 사용하더라도 운율을 학습하기 위한 F0의 경우의 수는 충분하므로, 딥러닝 기반 모델인 경우에 높은 자연성을 나타내는 것으로 분석할 수 있다. 음절의 경계가 HTS 방식 합성음은 부자연스러울 때가 있고, Tacotron 기반 합성음은 명확하지 않을 때가 있는데, 이 또한 Tacotron 기반 합성음의 명료도를 낮추고, HTS 방식 합성음의 자연성을 낮추는 요인으로 추측된다. 소스-필터 모델 기반의 보코더로 인한 특유의 윙윙거리는 소리와 쉰 소리는 Tacotron 기반 합성음에서는 나타나지 않는다.

3.3. 분석

TTS 시스템의 경우, 아직까지도 MOS 평가를 대신할 만한 객관적 음질 평가가 존재하지 않으며, 따라서 다양한 실험 조건에 따른 비교 결과를 객관적으로 나타내는 데 어려움이 있다. 그러나 다행히도 attention 메커니즘 기반의 모델에서는 alignment 그래프를 통해 그 모델의 성능을 시각적으로 확인할 수 있다. Alignment 모델이 디코더가 제대로 된 입력을 기반으로 디코딩을 하도록 인도하는 역할뿐 아니라, 화자의 특성을 담아낸 duration 모델의 역할까지 하기 때문이다. 만약 alignment가 끊기거나 반복되면 그대로 합성음이 출력되기 때문에 심각한 문제를 초래하게 된다. Alignment 그래프의 가로축은 입력 문자열의 몇 번째 문자인지를 나타내고, 세로축은 디코더의 time step을 나타낸다. Alignment가 묵음이 아닌 구간에서 잘 이어지면서 선명하고 값이 클수록 모델의 성능이 좋다고 할 수 있다.

3.3.1. 훈련 데이터의 구성이 alignment에 미치는 영향

Alignment에 영향을 미치는 첫 번째 요인은 발음 자체였다. 자음인데도 모음과 같이 울림소리인 'ㅁ' 받침과 'ㄹ'은 합성음에서 명확하게 발음이 되지 않았고, 문장 안에서 비슷한 발음이 연달아 나오는 경우, 앞선 발음이 반복되거나 중간이 끊기기도 했다.

또한 훈련 데이터의 길이나 문장의 형식 등의 구성이 균형 잡혀 있지 않기 때문에, 훈련 데이터 내의 비중이 적은 문장의 경우 alignment의 일반화가 잘 이루어지지 않는 문제점이 존재했다. 아주 짧거나 아주 긴 문장의 비율이 적기 때문에, 짧거나 긴 문장을 합성할 때는 alignment 중 일부가 반복되거나 끊기는 현상이 나타났다. 또한 문장부호를 포함시킨 채로 훈련을 시키는 상황에서 훈련 데이터의 대부분의 문장이 마침표로 끝나기 때

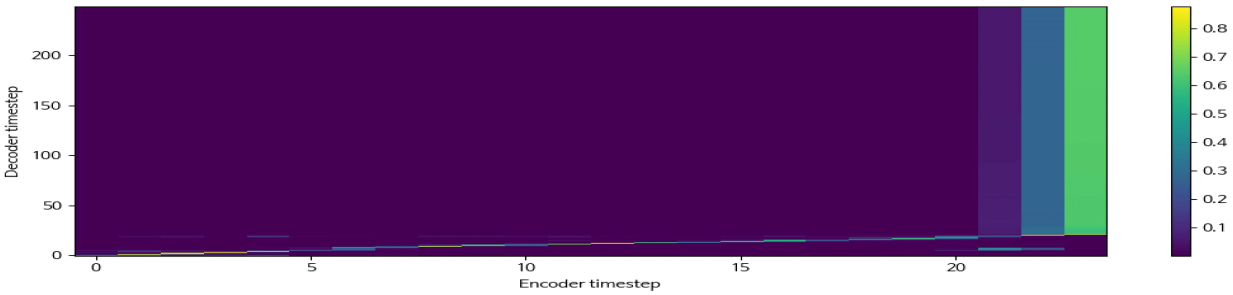

그림 3. 문장부호가 포함된 문장의 alignment 예. '첫째, 도망치는 거다.'
Figure 3. An example of a sate containing punctuation. '첫째, 도망치는 거다.'

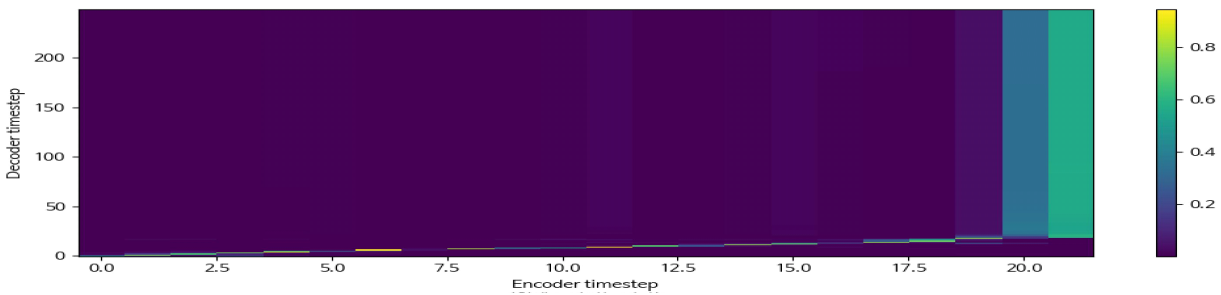

그림 4. 문장부호가 포함되지 않은 문장의 alignment 예. '첫째 도망치는 거다.'
Figure 4. An example of a sate without punctuation. '첫째 도망치는 거다.'

문에 마침표로 끝나지 않는 문장은 끝부분의 alignment의 값이 보다 작거나 반복되는 현상이 나타났다. 비슷한 현상으로, '-다.'로 끝나지 않는 문장은 평균적으로 '-다.'로 끝나는 문장에 비해 alignment 값이 작게 나타나는 경향이 있었다.

<그림 3>과 <그림 4>는 각각 문장부호가 포함된 문장 '첫째, 도망치는 거다.'와 문장부호가 포함되지 않은 문장 '첫째 도망치는 거다'를 합성했을 때의 alignment를 나타낸다. <그림 3>에서 가로축 22번째가 '.', 23번째가 EOS이다. 문장 중간의 쉼표에서 마침표에서와 같이 묵음이 나타나기 때문에 마침표에 집중하는 Decoder timestep이 두 번 나타나는 것을 확인할 수 있다. <그림 4>에서는 가로축 20번째가 'ㅏ', 21번째가 EOS이다. 'ㅏ'에 집중해야 하는 Decoder timestep에서 'ㅏ'에 집중하지 못하고 EOS에 집중함으로 인해 끝부분이 선명하지 않게 합성되었다. 이후의 결과들도 모두 alignment 및 청취 성능을 기준으로 분석하였다.

3.3.2. 훈련 데이터의 양에 따른 인코더 설정

실험 결과를 통해, 훈련 데이터의 양이 적을수록 인코더 CBHG의 K, 즉 모델링하는 *n*-gram의 최대 *n*이 작아야 한다는 것을 발견했다. 훈련 데이터양이 많을 때보다 적을 때 학습이 가능한 대 n의 범위도 좁아졌다. 이러한 현상은 언어에 상관없이 나타났고, 훈련 데이터의 양이 적을수록 문맥의 양이 기하급수적으로 줄어들기 때문이라고 추측할 수 있다. 표 3과 4는 입력 문자, 출력 언어 쌍이 각각 알파벳, 영어일 때와 한글, 한국어일 때 DB 양에 따른 최적의 인코더 CBHG의 K를 보여준다.

표 3. DB 양에 따른 최적의 인코더 CBHG의 K
(문자로 알파벳을 사용한 영어음성 합성)
Table 3. The best value of K from encoder CBHG
(English TTS using alphabets)

DB 양(시간)	11.62	21.04
인코더 CBHG K	5	16

CBHG, 1-D convolution bank + highway network + bidirectional gated recurrent unit; TTS, text-to-speech

표 4. DB 양에 따른 최적의 인코더 CBHG의 K
(문자로 한글을 사용한 한국어음성 합성)
Table 4. The best value of K from encoder CBHG
(Korean TTS using Hangeuls)

DB 양(시간)	6.31	9.04
인코더 CBHG K	3	5

CBHG, 1-D convolution bank + highway network + bidirectional gated recurrent unit; TTS, text-to-speech

3.3.3. 합성음의 언어와 사용하는 문자에 따른 후처리 네트워크 설정

또한 한글을 문자로 사용한 한국어 TTS 시스템의 경우, 후처리 네트워크로서 highway 네트워크를 사용해야 CBHG를 사용했을 때와 달리 alignment가 선명한 직선 형태로 수렴하며 큰 값을 가졌고, 실질적인 합성음 출력이 가능했다. 같은 훈련 데이터를 사용한 알파벳을 문자로 사용한 한국어 TTS 시스템의 경우에는 CBHG와 highway 네트워크를 후처리 네트워크로 이용했을 때 합성음끼리의 청취 성능에는 큰 차이가 없었다. 그러나 alignment는 highway 네트워크를 사용할 때가 조금 더 수렴이 잘 되었다. 알파벳을 문자로 사용한 영어 TTS 시스템의 경우, CBHG와 highway 네트워크 모두 사용 가능했으나, CBHG를 사용하는 것이 더 좋은 청취 성능을 보였다. 청취 성능 외에도 검증 데이터의 loss 및 멜 스케일 loss와 선형 스케일 loss의 차이를 통해 성능 차이를 확인할 수 있었다.

이를 분석해보면 우선, 같은 훈련 데이터로 동일 태스크를 수행할 때 사용하는 문자가 알파벳일 때보다 한글일 때 후처리 네트워크로 인한 차이가 큰 것으로 보아, 사용하는 문자의 영향을 받는다는 것을 알 수 있다. 영어 TTS 시스템의 경우와는 훈련 데이터의 양, 화자, 합성하는 언어가 모두 다르기 때문에 하나의 결론을 유추하기는 어렵다. 따라서 동일한 영어 훈련 데이터를 양만 기존 데이터의 절반으로 줄여서 실험을 수행했고, 이를 통해 CBHG보다 highway 네트워크가 후처리 네트워크로서 적합하다는 결과를 얻었다. 결론적으로 합성하는 언어와 사용하는 문자 그리고 훈련 데이터양에 따라 문맥의 양이 결정되고, 문맥의 양이 많을 때와는 달리 그 양이 적을수록 후처리 네트워크가 복잡하면 alignment 모델을 훈련시키기 어렵다는 것으로 분석할 수 있다.

3.3.4. 구조에 따른 성능 분석

CBHG에서의 residual connection과 max pooling이 모델의 수렴 및 일반화를 돕는다는 것을 실험적으로 확인하였다. 마찬가지로 실험을 통해 확인한 결과, CBHG의 highway 네트워크의 층수는 4개 혹은 1개보다 2개가 모델의 수렴 및 일반화를 도왔다. 이 외에도 CBHG의 1차 convolution bank의 채널 수는 128개보다 64개가, 인코더 및 디코더의 pre-net의 첫 번째 층의 노드 개수는 256개보다 128개가, 문자 임베딩의 차원은 256차보다 128차가, 디코더의 RNN 종류는 Hochreiter & Schmidhuber(1997)의 LSTM이나 Collins *et al*.(2017)의 UGRNN(Update Gate Recurrent Neural Network)보다는 GRU가 도움이 되었다.

3.3.5. 디코더의 표적에 따른 성능 분석

각각 40, 80, 160 밴드의 멜 스펙트로그램을 디코더의 표적으로 설정하여 비교한 결과, 밴드 개수가 40인 스펙트로그램보다 80 혹은 160인 스펙트로그램일 때 alignment가 더 잘 되고 합성음의 청취 성능도 좋았다. 밴드 개수가 80일 때와 160일 때는 큰 차이가 없어 80 밴드의 멜 스케일 스펙트로그램을 디코더의 표적으로 설정했다.

또한 Wang *et al*.(2017)이 Tacotron에서의 후처리 네트워크의 효능을 보일 때 제안한 대로 멜 스케일과 선형 스케일 스펙트로그램의 loss를 모두 훈련에 사용한 모델과 디코더의 표적으로 바로 선형 스케일 스펙트로그램을 설정하면서 후처리 네트워크를 제외한 모델을 비교했는데, 전자의 성능이 높은 이유가 멜 스케일 loss도 훈련에 사용해서인지, 단순히 모델이 더 깊어져

서인지는 명확하지 않다. 따라서 후처리 네트워크는 동일하게 사용하되, 멜 스케일 loss를 제외하고 선형 스케일 loss만 훈련에 사용한 모델과 멜 스케일과 선형 스케일 loss를 모두 훈련에 사용한 모델을 비교해봤다. 그 결과, 제안한 대로 멜 스케일 loss도 훈련에 사용하는 것이 더 성능이 좋았다. 나아가 멜 스케일 스펙트로그램이 그 자체로 훈련에 도움이 되는 것인지, 차원이 작아서 훈련이 잘 되는 것인지를 확인하기 위해 후처리 네트워크를 제외하고, 멜 스케일 스펙트로그램과 선형 스케일 스펙트로그램을 concatenate한 것을 디코더의 표적으로 설정하여 실험을 해봤다. 제안된 모델과 비교했을 때 alignment의 수렴이 대체로 비슷하고 짧은 문장의 경우는 대체로 끝이 반복되는 현상이 없어 더 잘 학습되었다. 디코더의 표적이 선형 스케일 스펙트로그램보다도 차원이 크고 전체 모델의 깊이가 얕아진 것을 감안하면 멜 스케일 스펙트로그램이 성능 개선에 확실히 기여한다고 할 수 있다. 반면, 목소리가 실제보다 조금 더 높고 음성의 명료도가 낮았다. 이는 스펙트로그램에서 중간 주파수 범위가 잘 학습되지 않아 발생한 것으로 예상된다.

스펙트로그램을 구할 때 사용한 프레임 길이와 오버랩 길이는 각각 100 ms, 25 ms로, reduction factor는 4 혹은 5로 최적의 하이퍼파라미터를 찾아 설정했다. 이러한 실험들을 토대로 효율적으로 음성의 정보를 나타내는 표적을 찾는 것이 중요함을 확인했다.

4. 결론

본 연구는 Tacotron에 기반한 end-to-end 한국어 TTS 시스템을 구현하고 분석하였다. End-to-end 합성 방식은 기존의 방식과 달리 텍스트 분석부, 음향 모델링부, 음성 합성부에 대해 전문적인 지식 없이도 구현이 가능하기 때문에, 사람의 경험을 토대로 왜곡할 수 있다는 문제가 없고 진입장벽이 낮으며, 각 부분에서의 loss가 쌓이지 않아 효율적인 훈련이 가능하다. 더불어 텍스트 분석부를 사용하지 않기 때문에, 다양한 언어 음성 합성에 적용이 용이하며, 평소에 거의 쓰지 않는 낯선 문자의 조합도 모델링이 가능하다.

본 연구에서는 구글이 사용한 훈련 데이터의 37% 분량의 적은 양의 훈련 데이터를 사용해서 MOS 2.98, DMOS 3.25의 자연성이 높은 한국어 합성음을 출력했다. 이러한 과정에서 기여한 점은 적은 양의 훈련 데이터를 사용할 때는 인코더 CBHG에서 모델링하는 *n*-gram의 최대 *n*도 작아야 한다는 사실을 발견했다는 점과 후처리 네트워크로 highway 네트워크를 사용하여 사용한 훈련 데이터에 대해 한국어 TTS 시스템을 가능하게 했다는 점이다. 마지막으로, 모델이 수렴 및 일반화를 할 수 있도록 적절한 디코더의 표적을 설정하는 것이 중요함을 확인했다.

한편, end-to-end 시스템은 훈련 데이터의 양과 구성에 영향을 많이 받기에 훈련 데이터에 비해 너무 짧거나 긴 문장을 합성할 때를 비롯한 몇 가지 경우에는 alignment에 있어 어려움을 겪었다. 따라서 alignment 모델의 개선이 가장 근본적인 향후 연구 방향이고, 다음과 같은 계획들이 있다. 우선 더 많은 양의 훈련 데이터를 수집하여 사용해야 하며, Collins *et al*.(2017)이 밝힌 바에 따르면 RNN 구조 자체의 성능은 어느 정도 수렴하므로 짧은 훈련 데이터 후에 긴 훈련 데이터를 학습시키는 등의 훈련 방식을 다양하게 적용해보는 것이 효과적일 것이다(Bengio *et al*., 2009). 또한 Deep Voice 3에서와 같이 디코더 time step이 커질 때 디코더가 집중하는 인코더 time step이 작아지지 않도록 alignment를 훈련시키는 monotonic attention 메커니즘을 현재의 attention 메커니즘 대신 사용해 볼 수 있다(Raffel *et al*., 2017). 한편, 다중 화자의 임베딩 값을 활용하면 하나의 모델로 여러 화자의 음성을 합성할 수 있으며, 여러 화자의 데이터를 통해 각 화자의 alignment 정보를 배우는 데에도 도움이 된다. 나아가 한 화자의 alignment 정보를 다른 화자의 alignment 대신 적용하면 원하는 대로 억양 등 화자의 발화 특성을 바꾸는 것도 가능하다. 화자 임베딩처럼 감정 임베딩 값을 활용하면 하나의 시스템으로 다양한 감정을 표현하도록 음성을 합성할 수도 있다.

이 외에도 음성 합성에 사용한 Griffin-Lim 알고리즘으로 인해 합성음의 명료도가 낮았고, Ping *et al*.(2017)과 Shen *et al*.(2017)의 결과에서 알 수 있듯이 이를 WaveNet으로 대체하면 음질을 개선할 것으로 예상되지만, 훈련 시간이 급격하게 늘어날 것이므로 한정된 자원으로 할 수 있는 다른 방법을 찾는 것이 필요하다. 또한 텍스트 정규화 과정 중 아라비아 숫자를 한글로 바꾸는 알고리즘의 정확도가 97.2%였고, 이를 개선하면 역시 alignment 모델의 수렴에 도움이 될 것으로 예상된다.

참고문헌

Arik, S., Chrzanowski, M., Coates, A., Diamos, G., Gibiansky, A., Kang, Y., Li, X., Miller, J., Ng, A., Raiman, J., Sengupta, S., & Shoeybi, M. (2017a). Deep Voice: Real-time neural text-to-speech. *Proceedings of the 34th International Conference on Machine Learning* (pp. 195-204). Sydney, AU. 6-11 August, 2017.

Arik, S., Diamos, G., Gibiansky, A., Miller, J., Peng, K., Ping, W., Raiman, J., & Zhou, Y. (2017b). Deep Voice 2: Multi-speaker neural text-to-speech. *Advances in Neural Information Processing Systems 30* (pp. 2966-2974). Long Beach, CA. 4-9 December, 2017.

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. Retrieved from http://arxiv.org/abs/1409.0473 [Computing Research Repository] on January 9, 2018.

Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). Curriculum learning. *Proceedings of the 26th Annual International Conference on Machine Learning* (pp. 41-48). 14-18 June, 2009.

Cho, K., Van Mrriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. Retrieved from http://arxiv.org/abs/1406.1078 [Computing

Research Repository] on January 9, 2018.

Chung, J., Gulçehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. Retrieved from http://arxiv.org/abs/1412.3555 [Computing Research Repository] on January 9, 2018.

Collins, J., Sohl-Dickstein, J., & Sussillo, D. (2017). Capacity and trainability in recurrent neural networks. *Proceedings of the 5th International Conference on Learning Representations*. Retrieved from https://openreview.net/forum?id=BydARw9ex on January 9, 2018.

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

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770-778). 26 June-1 July, 2016.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780.

Hunt, A. J., & Black, A. W. (1996). Unit selection in a concatenative speech synthesis system using a large speech database. *Proceedings of the 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing* (pp. 373-376). 7-10 May, 1996.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Proceedings of the 32nd International Conference on Machine Learning* (pp. 448-456). 2 Mar, 2015.

Kawahara, H. (1997). Speech representation and transformation using adaptive interpolation of weighted spectrum: Vocoder revisited. *Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing* (pp. 1303-1306). 21-24 April, 1997.

Lee, J., Cho, K., & Hoffman, T. (2016). Fully character-level neural machine translation without explicit segmentation. Retrieved from http://arxiv.org/abs/1610.03017 [Computing Research Repository] on January 9, 2018.

Morise, M., Yokomori, F., & Ozawa, K. (2016). WORLD: A vocoder-based high-quality speech synthesis system for real-time applications. *IEICE Transactions on Information and Systems*, 99(7), 1877-1884.

Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., & Kavukcuoglu, K. (2016). WaveNet: A generative model for raw audio. Retrieved from http://arxiv.org/abs/1609.03499 [Computing Research Repository] on January 9, 2018.

Ping, W., Peng, K., Gibiansky, A., Arik, S., Kannan, A., Narang, S., Raiman, J., & Miller, J. (2017). Deep voice 3: Scaling text-to-speech with convolutional sequence learning. Retrieved from http://arxiv.org/abs/1710.07654 [Computing Research Repository] on January 9, 2018.

Rabiner, L., & Schafer, R. (2011). *Theory and applications of digital speech processing*. New Jersey: Pearson.

Raffel, C., Luong, M.-T., Liu, P., Weiss, R., & Eck, D. (2017). Online and linear-time attention by enforcing monotonic alignments. *Proceedings of the 34th International Conference on Machine Learning* (pp. 2837-2846). 6-11 August, 2017.

Shen, J., Pang, R., Weiss, R., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerry-Ryan, R., Saurous, R., Agiomyrgiannakis, Y., & Wu, Y. (2017). Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions. Retrieved from http://arxiv.org/abs/1712.05884 [Computing Research Repository] on March 1, 2018.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1), 1929-1958.

Srivastava, R., Greef, K., & Schmidhuber, J. (2015). Highway networks. Retrieved from http://arxiv.org/abs/1505.00387 [Computing Research Repository] on January 9, 2018.

Sutskever, I., Vinyals, O., & Le, Q. (2014). Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems 27* (pp. 3104-3112). 8-13 December, 2014.

Tokuda, K., Nankaku, Y., Toda, T., Zen, H., Yamagishi, J., & Oura, K. (2013). Speech synthesis based on hidden markov models. *Proceedings of IEEE*, 101(5), 1234-1252.

Vinyals, O., Kaiser, Ł., Koo, T., Petrov, S., Sutskever, I., & Hinton, G. (2015). Grammar as a foreign language. *Advances in Neural Information Processing Systems 28* (pp. 2773-2781). 7-12 December, 2015.

Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., Le, Q., Agiomyrgiannakis, Y., Clark, R., & Saurous, R. (2017). Tacotron: Towards end-to-end speech synthesis. Retrieved from http://arxiv.org/abs/1703.10135 [Computing Research Repository] on January 9, 2018.

Wu, Z., Watts, O., & King, S. (2016). Merlin: An open source neural network speech synthesis system. *Proceedings of the 9th ISCA Speech Synthesis Workshop* (pp. 218-223). Sunnyvale, CA. 13-15 September, 2016.

• **최연주 (Choi, Yeunju)**
한국과학기술원 전기및전자공학부
대전광역시 유성구 대학로 291
Tel: 042-350-7617
Email: wkadldppdy@kaist.ac.kr
관심분야: 음성합성
현재 전기및전자공학부 박사과정 재학 중

• **정영문 (Jung, Youngmoon)**
한국과학기술원 전기및전자공학부
대전광역시 유성구 대학로 291
Tel: 042-350-7617
Email: dudans@kaist.ac.kr
관심분야: 음성 검출
현재 전기및전자공학부 박사과정 재학 중

• **김영관 (Kim, Younggwan)**
한국과학기술원 전기및전자공학부
대전광역시 유성구 대학로 291
Tel: 042-350-7617
Email: cleanthink@kaist.ac.kr
관심분야: 음성인식, 화자적응
현재 전기및전자공학부 박사과정 재학 중

• **서영주 (Suh, Youngjoo)**
한국과학기술원 전기및전자공학부
대전광역시 유성구 대학로 291
Tel: 042-350-7517 Fax: 042-350-7619
Email: yjsuh@kaist.ac.kr
관심분야: 음성합성, 음성신호처리
2006~현재 전기및전자공학부 연구교수

• **김회린 (Kim, Hoirin)** 교신저자
한국과학기술원 전기및전자공학부
대전광역시 유성구 대학로 291
Tel: 042-350-7417 Fax: 042-350-7619
Email: hoirkim@kaist.ac.kr
관심분야: 음성인식, 화자인식, 패턴인식
2001~현재 전기및전자공학부 교수