음성합성을 위한 딥러닝 기반 생성모델

엔씨소프트 ▌양진혁

1. 서 론

최근 딥러닝 기술은 영상, 음성, 자연어 분야의 생성모델들이 높은 품질의 결과물들을 생성할 수 있게 하며 많은 연구가 이루어지고 있다 [1-4]. 이러한 딥러닝 기술은 흔히 TTS (Text-To-Speech)로 알려진 음성합성 분야에서도 좋은 성과를 내고 있다.

전통적으로 음성합성을 위해 보편적으로 사용된 기술은 연결 합성 (concatenative synthesis) 방법과 통계기반 파라미터 합성 (statistical parametric speech synthesis) 방법이다. 대표적인 연결 합성 방법인 음편접합방법론 (unit selection synthesis) [5]은 음소나 단어 같은 음성 신호의 작은 단위인 음편(speech unit)으로 데이터베이스(Database, DB)를 구축하고, 합성 시에 입력 문장을 토대로 연결될 최적의 음편을 선택하여 이어 붙인다. 이 방법은 합성 문장이 DB의 음성과 유사할수록 좋은 성능을 보이지만, 그렇지 않은 문장은 접합부가 부자연스럽고, 음성을 다양하게 만들기 어렵게 한다. 또한, 이 방식은 기본적으로 데이터가 매우 많이 필요하다. 통계기반 파라미터 합성 방법은 은닉 마르코프 모델 (Hidden Markov Model, HMM)을 사용하여 음성 신호 처리 기법으로부터 얻은 파라미터를 추정하도록 학습한다 [6, 7]. 이 방법은 음성의 특성을 바꿀 수 있는 등 유연성이 높으나, 합성음의 품질이 좋지 않다는 한계가 있다.

최근 딥러닝 기술이 도입되기 시작하면서 음성합성 기술은 비약적으로 발전했다. 딥러닝은 통계기반 파라미터 합성 방법과 결합하여 점진적으로 기술을 발전시켜오다가 WaveNet [3], SampleRNN [8], Char2Wav [9], Tacotron [10] 연구들에 이르러 텍스트로부터 음성신호를 직접적으로 예측할 수 있게 됨으로써 성능은 비약적으로 높이면서 음성합성 분야에 대한 진입장벽은 낮추었다. 현재 딥러닝 기반의 음성합성 모델들은 사람이 직접 녹음한 음성과 유사한 수준의 음질과 자연스러움을 달성하고 있다. 또한, 딥러닝 기술은 더 나아가 다양한 스타일의 음성을 표현하고, 적은 데이터로도 다수의 화자에 대해 높은 음질을 달성하는 등 도전적인 연구 주제들도 계속해서 다루고 있다.

이 논문에서는 음성합성 분야에서 딥러닝 기반 생성모델들이 어떻게 적용되고 발전되고 있는지 최신 연구들까지 소개한다. 논문은 다음과 같이 구성된다. 섹션 2에서는 음성합성 구성요소 기술을 구분하여 소개한다. 섹션 3과 4에서는 각각 구성요소인 딥러닝 기반의 TTS 모델과 보코더 모델 분야에서 제기된 문제들과 이를 해결하기 위해 제안된 방법들을 소개한다. 섹션 5에서 논문을 마무리한다.

2. 구성 요소 기술

딥러닝 기반의 음성합성 시스템은 크게 두 가지의 모델로 구성된다. 첫번째는 입력 텍스트를 음향 특징(acoustic feature)으로 변환하는 TTS 모델이고, 두번째는 음향 특징을 음성 웨이브폼(waveform)으로 변환하는 보코더 모델이다. 일반적으로 음성합성 시스템 전체를 TTS 모델이라고 하는 경우도 있으나, 텍스트를 음향 특징으로 변환하는 모델만을 지칭하기 위해 협의된 명시적인 표현이 없으므로 이 논문에서는 해당 모델을 TTS 모델이라고 지칭한다.

이 섹션에서는 두 모델을 연결하는 대표적인 중간 특징 역할인 멜 스펙트로그램을 소개하고, TTS 모델과 보코더 모델이 각각 어떤 역할을 하는지 설명한다.

2.1 멜 스펙트로그램 (Mel Spectrogram)

TTS 모델의 출력이면서 보코더 모델의 입력이 되는 음향 특징은 연구자에 따라 다양하게 사용한다. 멜 스펙트로그램은 그 중에서도 주요 딥러닝 기반 음성합성 연구에서 가장 일반적으로 사용하는 음향 특징이다. 음향 특징을 사용하여 두 단계에 걸쳐서 음성을 생성하는 대표적인 이유는 첫째로 음성 신호를 유의미하고 분석하기 쉬운 형태로 가공하기 위함이고, 둘째로 매우 긴 시계열 데이터인 음성 신호를 시간 축

으로 부담을 줄여주기 때문이다. 이러한 요소들은 딥러닝 모델의 학습을 용이하게 한다. 특히 디지털 오디오를 위해 주로 사용되는 샘플링 주파수 (sampling frequency)는 44,100 헤르츠(Hz)인데, 이는 10초 길이의 음성을 위해 모델이 약 44만 개의 샘플을 생성해야 한다는 뜻이다. 이렇게 긴 길이의 시계열 데이터는 장기 의존성 (long-term dependency) 문제와 높은 연산량을 요구하는 등 딥러닝 모델에서 널리 알려진 이슈를 갖는다 [11].

음성 신호로부터 멜 스펙트로그램을 추출하기 위한 절차는 다음과 같다. 먼저, 짧은 시간 단위로 나누어 푸리에 변환을 하는 단시간 푸리에 변환을 적용한다. 이 때, 특정한 샘플만큼 건너 뛰며 윈도우 함수를 적용하여 음성 신호의 길이가 건너 뛰는 길이의 배수만큼 축소된다. 이를 통해 시간 정보와 주파수 정보를 모두 갖는 스펙트로그램을 추출할 수 있다. 이 스펙트로그램의 진폭값에 절대값과 로그를 취하여 데시벨(decibel, dB)로 변환한다. 또한, 변환된 스펙트로그램의 주파수 축으로 멜 스케일(mel scale)로 변환하면 비로소 멜 스펙트로그램을 얻는다. 로그와 멜 스케일을 적용하는 것은 사람의 청각기관의 특성을 고려한 것이다. 최종적으로 만들어진 멜 스펙트로그램은 일반적으로 한 프레임당 80차원의 벡터이며, 웨이브폼의 1/256배의 시간 길이를 갖는다 [10]. 이렇게 만들어진 멜 스펙트로그램은 음성을 분석하기 위해 필요한 정보를 압축하여 갖고 있으며, 그 활용도가 매우 높다. 이 논문에서는 중간 역할을 하는 음향 특징으로써 멜 스펙트로그램을 기준으로 연구들을 소개한다.

2.2 TTS 모델

TTS 모델은 텍스트를 입력으로 사용해서 멜 스펙트로그램(혹은 다른 음향 특징)을 생성하는 모델이다. 입력으로 텍스트가 들어오면 철자 단위의 임베딩 벡터로 변환하여 사용하며, 이를 철자 임베딩 (character embedding)이라고 한다. 한국어를 학습하기 위해서는 입력 문자를 초성, 중성, 종성으로 나누어 사용하며, 영어를 학습하기 위해서는 알파벳으로 나누어 사용한다. 또한, 성능을 높이기 위해 음소 (phoneme) 표기로 변환하여 사용하기도 한다. 출력으로는 멜 스펙트로그램을 정규화 (normalize)하여 사용하며, 그 길이가 입력 텍스트의 길이와 동일하지 않고 동적으로 생성된다.

따라서, 딥러닝 모델의 엔드투엔드 (end-to-end) 학습을 위해서는 입력과 출력의 관계를 매핑하는 모듈이 필요한데, 어텐션 (attention mechanism) [12]이 이를 가능하게 한다. 이러한 특성은 자연어 처리분야, 특히 기계번역과 많은 부분에서 유사하며 실제로 다수의 TTS 모델 연구가 신경망 기반 기계번역 (Neural Machine Translation, NMT) 연구의 기술들을 도입하였다.

TTS 모델의 출력인 멜 스펙트로그램은 음성 신호의 주요한 정보를 갖고 있다. 이는 TTS 모델 단에서 음성의 특성 (예를 들면, 높낮이, 억양, 화자 정체성 등)을 대부분 결정한다는 것을 뜻한다. 따라서, TTS 모델 연구로써 텍스트를 음향 특징으로 변환하는 것과 더불어, 다화자 음성합성 연구, 음성 스타일 조정 연구 등도 이루어지고 있다.

2.3 보코더 모델

보코더 모델은 TTS 모델을 통해 추정된 멜 스펙트로그램을 음성 웨이브폼으로 변환하는 모델이다. 음성을 직접적으로 청취할 수 있는 형태는 웨이브폼이기 때문에 보코더 모델은 깨끗하고 명료한 음질로 음성을 제공하기 위한 가장 주요한 역할을 한다. 또한, 음성의 특성을 결정하는 정보는 TTS 모델이 텍스트로부터 멜 스펙트로그램을 생성하면서 대부분 결정되었기 때문에, 보코더 모델은 멜 스펙트로그램의 정보를 기반으로 음성 신호를 왜곡하지 않고 변환하는 것도 중요하다. 이 과정에서 보코더 모델은 적게는 수만 개에서 많게는 수십만 개의 샘플을 생성하기 때문에 생성 속도도 주요한 이슈이다.

TTS 모델은 출력의 길이가 동적이기 때문에 어텐션이 필요했던 것에 반해, 보코더 모델의 입력과 출력은 그 길이가 일정하게 비례하므로 별도의 장치가 필요하지 않다.

3. 딥러닝 기반 TTS 모델

이 섹션에서는 딥러닝 기반 TTS 모델 연구가 어떻게 진행되었는지 최신연구들을 소개한다.

3.1 자기회귀 TTS 모델 (autoregressive TTS model)

초기 딥러닝 기반 TTS 모델 연구는 자기회귀 모델을 기반으로 하고 있으며, NMT 분야의 언어 모델(language model)기반 딥러닝 연구들을 도입하고 음성합성에 맞게 응용한 연구들이 주를 이룬다. 딥러닝 기반 TTS 모델에서 출력인 음성 신호의 결합확률은 텍스트 정보와 특정 시간 전까지의 음성 신호가 주어졌을 때 다음 음성신호가 나올 조건부 확률의 곱으로 나타낼 수 있으며, 아래와 같이 정의된다.

$$p(\mathbf{s} \mid \mathbf{h}) = \prod_{t=1}^{T} p(s_t | s_1, s_2, \ldots, s_{t-1}, \mathbf{h}),$$

여기서 s_t는 멜 스펙트로그램 혹은 스펙트로그램의 t번째 프레임에 해당하고, $\mathbf{h}$는 텍스트이다.

Tacotron [10]은 TTS 모델을 위해 엔드투엔드 학습을 적용한 시발점이 되는 연구이다. 모델은 어텐션을 사용한 순환신경망 (Recurrent Neural Network, RNN) [12] 구조이다. 이 연구에서는 아직 딥러닝 기반 보코더를 사용하지 않고, 전통적인 보코더인 그리핀-림 (Griffin-Lim) [13]을 사용한다. 따라서, 최종 출력으로 스펙트로그램(spectrogram)을 생성하도록 학습한다. 그리고나서 그리핀-림 알고리즘을 사용하여 스펙트로그램을 최종적인 웨이브폼으로 변환한다. TTS 모델의 성능을 측정하기 위해서는 주로 사용자들에게 샘플을 듣고 1점부터 5점까지 설문하는 평균 의견 점수 (Mean Opinion Score, MOS)를 사용한다. 이 연구는 비록 연결 방법보다 MOS가 다소 부족했으나, 기존의 복잡했던 음성합성 모델을 딥러닝을 적용하여 단순화했다는 점에서 의의가 있다.

Tacotron은 RNN 구조이기 때문에 학습에 소요되는 시간이 길다는 단점이 있었는데, 이를 보완하기 위해 RNN을 컨볼루션 신경망 (Convolutional Neural Network, CNN)으로 대체하고 개량한 DCTTS (Deep Convolutional TTS) [14]가 제안되었다. 또한, 저자는 학습 속도를 가속화하고 어텐션을 안정화하기 위해 유도 어텐션 손실함수 (guided attention loss)를 제안하였다. 유도 어텐션 손실함수는 인코더와 디코더를 연결하는 어텐션에 가중치를 주고 학습시키는 방법이다. 저자는 어텐션이 대각선 (diagonal)에 가까워지도록 가중치를 주었다. 이 부분에서 NMT와 TTS 모델의 어텐션에 대한 분명한 차이가 드러나는데, NMT는 번역 과정에서 단어의 위치가 출력할 때 역전된 순서로 나타날 수 있지만 TTS 모델은 들어온 순서

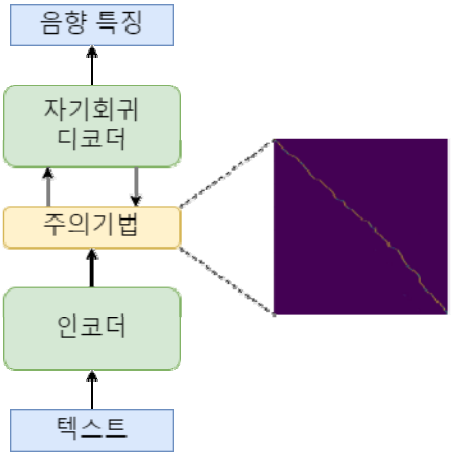

그림 1 자기회귀 TTS 모델 구조

대로 철자 혹은 음소에 대응한 음성신호가 나타난다. TTS 모델에서 어텐션은 매우 중요한 역할을 담당하는데, 어텐션의 가중치에 따라 음성의 길이에 영향을 주고 생략, 반복, 오발음 등의 문제가 발생할 수 있다. 이러한 방법들을 통해 Tacotron의 학습 시간으로 약 2주를 소요했던 것에 반하여 DCTTS는 2~3일만에 학습하고 더 높은 MOS를 달성하였다.

DCTTS는 모델 학습 시간을 단축하고 Tacotron보다 높은 MOS도 달성하였으나, 여전히 녹음된 음성과는 상당한 음질 차이가 있었다. 이를 위해 딥러닝 기반 보코더를 사용하여 음질을 대폭 개선한 Tacotron 2 [4]가 제안되었다. Tacotron과 DCTTS에서는 전통적인 보코더인 그리핀-림 알고리즘을 사용하였으나, 이 연구에서는 딥러닝 기반 보코더인 WaveNet을 사용하여 음질을 개선했다. 이 과정에서 중간을 연결하는 특징으로 더 이상 스펙트로그램을 사용하지 않고, 멜 스펙트로그램만을 출력으로 사용해서 더 높은 성능을 보임으로써 시스템을 간소화했다. 또한, 기존의 Tacotron은 내용 기반 어텐션 (content-based attention)을 사용하기 때문에 어텐션이 역전할 수 있다는 단점이 있었는데, Tacotron 2는 위치에 민감한 어텐션 (location-sensitive attention) [15]을 사용하여 이를 보완했다. 이 방법은 어텐션을 계산할 때, 이전 생성 단계의 어텐션 가중치를 누적해서 사용함으로써 어텐션이 역전하는 것을 방지해준다. 이러한 방법들로 Tacotron 2의 음성은 4.526의 MOS를 보였는데, 이는 전통적인 방법인 연결 방법 음성의 4.166보다 높은 MOS였으며 심지어 실제 녹음한 음성의 4.582와 상당히 유사한 MOS를 달성하였다.

3.2 비자기회귀 TTS 모델 (non-autoregressive TTS model)

TTS 모델 연구가 활발하게 진행되면서 생성된 음성의 음질과 자연스러움이 매우 빠르게 향상되었다. 하지만 자기회귀 방식으로 계산되는 어텐션을 사용하면 근본적으로 발화 오류(생략, 반복, 오발음)를 제거하기 어렵다. 또한, 자기회귀 방식의 TTS 모델이 실시간 합성 (1초 길이의 음성을 1초 내에 합성)은 가능하지만 음성합성이 일반적으로 스트리밍 서비스가 아니기 때문에 생성 속도가 충분하지 않다. 이를 위해서는 더 빠른 속도가 필요한데, 자기회귀 방식의 TTS 모델은 음성의 생성 길이와 비례하여 반복 추론해야 하는 특성으로 인해 한계가 있다. 이런 한계를 극복하고자 디코더의 추론 단계마다 계산했던 어텐션을 사용하지 않고, 인코더의 시퀀스를 디코더의 길이에 맞춰 일괄적으로 확장하고 병렬적으로 추론하는 비자기

회귀 모델들이 제안되었다.

FastSpeech [16]는 그런 배경에서 제안된 첫번째 비자기회귀 TTS 모델이다. 이 모델은 인코더와 디코더 모두 순방향 (feed forward)의 트랜스포머 (Transformer) [17, 18] 구조로 이루어져 있다. 또한, 그림 2를 보면 인코더와 디코더 사이에 어텐션을 대신하여 두 네트워크를 연결하는 길이 조절기가 있다는 점이 자기회귀 TTS 모델과의 큰 차이점이다. 길이 조절기의 역할은 인코더를 거친 벡터들을 각각 상응하는 디코더 프레임의 개수만큼 복제하는 것이다. 복제하기 위해서는 인코더의 벡터 (문자 표현)가 디코더의 벡터(음향 특징 표현)에 각각 어떻게 상응하는 지를 알아야 하는데, 이 수치는 사전에 학습한 자기회귀 TTS 모델의 어텐션으로부터 얻어낸다. 자기회귀 TTS 모델을 충분히 학습하고 나면, 학습 데이터셋 각각에 대해 어텐션의 가중치를 알 수 있기 때문에 이를 이용하여 인코더와 디코더의 관계를 추출한다. 이렇게 추출한 목표 길이 (target duration)는 학습 시에 길이 조절기의 입력으로 사용된다. 또한, 생성 단계에서는 추정된 길이를 사용하기 위해서 목표 길이를 정답으로 하고 인코더 벡터를 입력으로 하여 길이 예측 모듈 (duration predictor)을 학습한다. 따라서, 길이 조절기를 사용하면 인코더에 상응하는 디코더의 입력 벡터를 일괄적으로 전달할 수 있기 때문에 멜 스펙트로그램 전체를 한 번의 추론으로 생성할 수 있다.

한 번의 추론으로 멜 스펙트로그램 전체를 생성할 수 있게 된 FastSpeech의 생성 속도는 자기회귀 TTS 모델에 비해 약 269배 향상되었다. 이는 보코더를 제외하면, 10초 길이의 음성 샘플을 만드는데 약 30 ms가 걸리는 것을 의미하며 스트리밍 없이도 실시간 서비스가 가능한 수준이다. 또한, 생성할 때 길이 예측

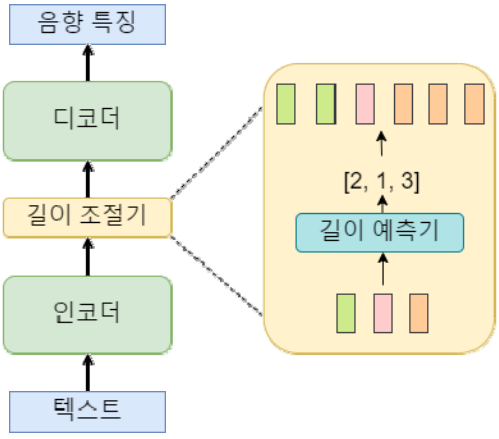

그림 2 비자기회귀 TTS 모델 구조 [16]

모듈이 추정한 인코더의 길이를 특정 배수를 곱해서 음성의 발화 속도를 효과적으로 조절하는 것이 가능하다. 추가 실험으로는 안정성 테스트 (반복, 생략 등)를 위해 TTS 시스템을 어렵게 하는 50문장을 선정하여 음성을 생성했는데, Tacotron 2가 24%의 오류율을 보인데 반해 FastSpeech는 0%를 달성하였다. 즉, 비자기회귀 TTS 모델 방식이 속도와 안정성에 있어 매우 효과적임을 보인 것이다.

속도와 안정성에서 FastSpeech가 좋은 성능을 보인 것에 반해, 아직 자기회귀 TTS 모델에 비해 MOS에서는 충분한 성능을 보이지 않았다. 여기에는 다양한 이유가 있겠지만, 대표적으로 일대다 매핑 문제 (one-to-many mapping problem) 때문이다. 일반적으로 텍스트와 음성의 관계를 볼 때, 입력 텍스트가 고정되어도 이에 대응하는 음성은 매우 다양하게 존재한다. 자기회귀 TTS 모델은 이전 단계까지의 정보를 이용하기 때문에 이 문제를 일부 해소할 수 있지만, 비자기회귀 TTS 모델은 그렇지 못하다. 이러한 문제를 해결하고자 디코더가 멜 스펙트로그램을 생성할 때 추가적인 정보를 사용하게 하는 FastSpeech 2 [19]가 제안되었다.

FastSpeech 2는 FastSpeech 모델을 기반으로 일대다 매핑 문제를 완화하도록 고안된 모델이다. FastSpeech 2는 정답 (ground-truth) 음성으로부터 피치 (pitch)와 에너지 (energy)를 추출하여 멜 스펙트로그램을 생성하기 전에 조건부 입력 (conditional input)으로 주고 이를 예측하도록 학습한다. 그림 3을 보면 멜 스펙트로그램과 피치의 관계를 볼 수 있는데, 멜 스펙트로그램에서 주름처럼 보이는 것들을 고조파 (harmonic)라고 하며, 피치 (실제로는 기본 주파수: 각주)와 정수배 관계에 있다. 또한, 에너지를 위해서 스펙트로그램의 L2-노름 (norm)을 사용한다. 이 특징들은 멜 스펙트로그램이 각 프레임에서 80차원의 벡터인 것과 달리 각 프레임에서 스칼라이다. 따라서, 피치와 에너지는 멜 스펙트로그램을 결정짓는 주요한 특징이면서 특정 프레임에서 각각 스칼라이기 때문에, 딥러닝 모델이 멜 스펙트로그램을 한 번에 생성하도록 학습하는 것에 비해 모델에 주는 부담을 상당히 완화할 수 있다.

이렇게 학습된 FastSpeech 2는 실험에서 3.83의 MOS를 달성하여 Tacotron 2의 3.70과 FastSpeech의 3.68보다 향상된 결과를 보였다. 또한 FastSpeech에서 길이 조절기를 이용해 발화 속도를 조절하였던 것처럼 조건부 입력으로 사용된 특징들을 조절하여 발화의 높낮이 등을 조절할 수 있게 되었다.

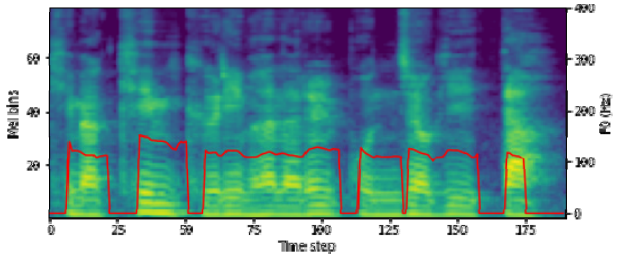

그림 3 멜 스펙트로그램과 기본 주파수 (피치)

4. 딥러닝 기반 보코더 모델

딥러닝 기반 보코더 연구에는 다양한 딥러닝 생성 모델이 사용되며, 대표적인 딥러닝 생성 모델의 세부적인 원리는 기존의 딥러닝 교재나 관련 논문들을 참고하기 바란다 [20-22].

딥러닝 기반 보코더 (neural vocoder)의 가장 대표적인 모델은 WaveNet [3]이다. 이 보코더 모델을 학습을 어렵게 하는 대표적인 이슈는 수십만개의 샘플을 생성해야 하고 그 과정에서 매우 넓은 수용 영역 (receptive field)이 필요하다는 것이다. 이를 위해 WaveNet에서는 팽창된 인과적 컨볼루션 (dilated causal convolution)를 쌓아 올린 구조를 도입한다. 이 구조는 상위 레이어의 팽창 값 (dilation)을 2의 제곱으로 늘려가며 깊게 쌓아서 효과적으로 수용 영역을 확장하고, 인과적 컨볼루션을 이용하여 자기회귀 과정에서 과거의 정보만을 이용하도록 한다. 모델은 다음 시간의 웨이브폼 값을 출력하고 로그-우도 (log-likelihood)를 최대화하도록 학습한다. 실험 결과로 언어 특징 (linguistic feature)을 조건부 입력으로 사용한 WaveNet이 통계기반 파라미터 합성 방법과 연결 방법보다 높은 MOS를 달성하였다.

하지만 WaveNet은 수십만개의 샘플을 자기회기 방식으로 반복적인 생성을 하기 때문에 생성 속도가 매우 느리다. 이를 위해 병렬적으로 생성할 수 있는 연구들이 진행되었다[23, 24]. WaveGlow [25]는 그 대표적인 모델로써, 이미지 생성 분야에서 생성 모델로 성공적인 결과를 보인 Glow [26]를 보코더 모델에 성공적으로 도입한 연구이다. Glow는 딥러닝 생성모델의 한 축인 플로우 기반 생성 모델 (flow-based generative model)이며, 이 방법은 입력 데이터의 분포를 학습하기 위해 역 변환(invertible transformation) 구조를 이용하고 음의 로그 우도(negative log likelihood)를 통해 학습한다. WaveGlow는 Glow 구조 내부에 WaveNet의 팽창 컨볼루션을 적용하고 멜 스펙트로그램을 업샘플링 (upsampling)하여 조건부 입력으로 사용함으로써 플로우 기반 생성 모델을 효과적으로 보코더 모델에 도입한다. 이러한 방법으로 WaveGlow는 생성 시에 더 이상 이전에 생성한 출력을 필요로 하지 않기 때문에 병렬적으로 웨이브폼을 생성할 수 있다. WaveNet이 초당 1~2백 개의 샘플을 생성하였는데, WaveGlow는 초당 52만 개의 샘플을 생성했다. 또한, WaveGlow는 MOS 실험에서 오픈 소스 WaveNet (각주)의 3.885보다 높은 3.961을 달성하였다.

하지만 WaveGlow는 파라미터 수가 많아 모델이 무겁고 학습을 위해 많은 GPU가 필요하다. 이러한 한계를 극복하고 성능을 개선하기 위해 적대적 신경망 (Generative Adversarial Network, GAN)기반의 보코더 모델들이 제안되기 시작했다.

MelGAN [27]은 멜 스펙트로그램을 직접적인 입력으로 사용하고 업샘플링과 팽창 컨볼루션을 포함한 얕은 모델로 웨이브폼을 생성하도록 생성기 (Generator)가 구성된다. 또한, 이미지 생성 모델로써 좋은 성능을 보였던 다중-스케일 판별기 (multi-scale discriminator) [28]와 특징 매칭 손실 함수(feature matching loss)를 도입한다. 이렇게 학습된 모델은 비록 WaveNet과 WaveGlow보다 낮은 MOS를 기록했지만 WaveGlow보다 10배 이상 빠른 생성 속도를 달성하였다.

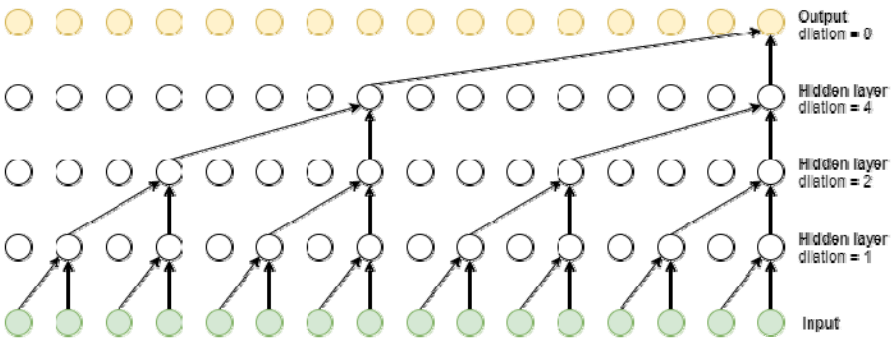

그림 4 팽창된 인과적 컨볼루션을 쌓아 올린 웨이브넷의 내부 구조. 출력 값이 갖는 수용 영역을 나타낸다. [3]

ParallelWaveGAN [29]은 인과적 컨볼루션을 사용하지 않는 WaveNet을 생성기로 사용하여 GAN 방법으로 학습한 모델이다. 또한, 손실함수로 다중-해상도 단시간 푸리에 변환 손실 함수 (multi-resolution STFT loss)를 함께 사용한다. 이 손실 함수는 해상도를 다르게 하는 다수의 인자를 사용하여 웨이브폼에 단시간 푸리에 변환을 적용한 후의 손실을 측정한 후 평균을 사용하며, 이는 모델이 음성의 시간-주파수 특성을 잘 배울 수 있게 한다. 실험 결과로 WaveNet이 MOS 3.61을 달성한데 반해, ParallelWaveGAN은 4.06으로 상당히 좋은 품질을 보였다.

VocGAN [30]은 MelGAN의 낮은 성능이 음성신호의 저주파와 고주파 특성을 충분히 학습하지 못한 데에서 비롯됐다고 판단한다. 따라서, 모델이 두 특성을 모두 직접적으로 학습할 수 있도록 계층적으로 중첩된 판별기 (hierarchically-nested discriminator) [31]와 공동 조건부 및 무조건부 판별기 (joint conditional and unconditional discriminator) [32]를 MelGAN의 구조에 추가로 적용한다. 이렇게 학습한 모델은 판별기와 손실함수 위주로 개선하였기 때문에 MelGAN만큼 빠르면서 ParallelWaveGAN보다 높은 성능을 보였다.

이러한 경향에 따라 최근에는 GAN 기반의 보코더 모델을 대부분 사용하고 있으며 [33], 속도와 품질 모두 서비스가 가능한 높은 수준을 보여주고 있다.

5. 이 슈

딥러닝 기반의 음성합성이 빠르게 연구되고 발전하고 있으며, 이와 함께 다양한 이슈들도 발생하고 있다. 그 중에서 주요한 이슈들을 소개한다.

우선, 음성합성은 다수의 사용자에게 설문 조사를 통해 얻어지는 MOS를 대표적인 평가 방법으로 사용한다. 특히 TTS 모델은 동일한 입력 텍스트에 매우 다양한 출력 특징이 가능하기 때문에 MOS에 높은 의존도를 보인다. MOS는 자연스러움 (Naturalness), 화자 유사도 (Speaker Similarity) 등으로 나누어서 평가하며 많은 문제를 안고 있다. 첫째, 청취자의 주관에 의존하는 평가이기 때문에 데이터와 모델이 동일해도 매 실험하다 다시 측정해야 한다. 즉, 모델의 성능평가를 위해서는 기존 방법이 이전 실험에서 MOS를 측정했더라도 모두 재현해서 새로 측정해야 한다. 둘째, 주관적인 평가이기 때문에 다수의 음성을 다수의 청취자에게 설문 받아야 하고, 이는 시간과 비용이 많이 소요된다. 마지막으로, 최근 음성합성 기술의 성능이 향상되면서 모델의 품질이 상향 평준화되어 일반적인 사람이 청취했을 때 성능의 차이를 체감하는 데에 한계가 생겼다. 자연스러움을 MOS로 측정하면 기존에는 운율의 어색함과 음질 등을 종합적으로 고려하여 모델의 품질을 구분하여 평가할 수 있었지만, 최신 모델이 생성한 음성은 점차 그 구분이 어려워지고 있다.

음성의 스타일을 학습하고 생성하는 것에 대한 문제가 있다. 이 논문에서는 음성합성 영역에서 딥러닝 생성모델이 어떻게 활용되고 있는지 설명하기 위해 주요한 연구 흐름 위주로 소개하였으나, 음성의 스타일을 학습하고 표현하는 주제들도 활발하게 연구되고 있다 [34, 35, 36]. 생성 모델은 주요 특성들에 대해 얽힘을 풀어내는 것 (disentanglement)과 조정 가능한 것 (controllability)이 중요하며, 이는 음성합성을 위한 모델도 마찬가지다. 음성을 더욱 실제처럼 생성하기 위해서는 감정, 호흡, 억양 등을 표현할 수 있어야 한다. 3.2에서 언급한 FastSpeech 2는 속도, 피치, 에너지 등이 조정 가능한데, 조정 요소의 선후 관계나 얽힘 문제로 품질 저하가 발생하기도 한다. 또한, 더 용이한 음성합성을 위해서는 감정과 같이 더 높은 단계의 표현 (high-level representation)이면서 사람이 인지하는 요소를 조정하는 것이 필요하다. 거기에 더해 청취자가 인지하기에 화자의 정체성 (speaker identity)은 일관되어야 한다.

이 외에도 다양한 활용을 위해 음성합성 분야에서는 초 단위의 음성 샘플로부터 화자의 특성을 높은 품질로 표현하는 연구 [37], 음성의 특성은 유지한 채 언어를 바꾸는 연구 [38], 외부 장치 없이 비자기회기 TTS를 학습하는 연구 [39, 40] 등 매우 다양한 연구들이 이루어지고 있다. 음성합성 분야는 딥러닝과 결합하며 전통적인 방법으로는 어려웠던 도전적인 주제들을 연구하고 있으며, 이를 위해 앞으로도 많은 연구들이 필요하다.

6. 결 론

딥러닝 기술은 음성 합성 분야에서 성공적으로 안착되고 있으며, 최신 모델로 생성한 음성들은 자연스러우면서도 좋은 음질을 보여주고 있다. 딥러닝 생성모델이 갖는 잠재력과 음성합성의 응용 분야를 고려하면 앞으로 그 중요성은 더 커질 것으로 기대된다.

본 논문에서는 딥러닝 기반 음성합성 모델을 구성하는 기본적인 구조를 설명하고, 이를 기준으로 최근까지의 TTS 모델과 보코더의 주요 흐름을 정리하였다. 그리고 일반적으로 딥러닝 음성합성 연구가 갖고 있는 이슈들을 소개했다. 앞으로 음성합성 연구를 시작하는 이들에게 작게나마 도움이 되기를 희망한다.

참고문헌

[1] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” CVPR, 2019.

[2] J. Shen, R. Pang, R. Weiss, M. Schuster, N. Jaitly, Z. Yang et al., “Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions,” ICASSP, 2018.

[3] A. V. D. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves et al., “Wavenet: A generative model for raw audio,” arXiv preprint arXiv:1609.03499, 2016.

[4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell et al., “Language models are few-shot learners,” arXiv preprint arXiv:2005.14165, 2020.

[5] A. J. Hunt and A. W. Black, “Unit selection in a concatenative speech synthesis system using a large speech database,” ICASSP, 1996.

[6] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, “Simultaneous modeling of spectrum, pitch and duration in hmm-based speech synthesis,” Sixth European Conference on Speech Communication and Technology, 1999.

[7] A. W. Black, H. Zen, and K. Tokuda, “Statistical parametric speech synthesis,” ICASSP, 2007.

[8] S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. Courville, and Y. Bengio, “Samplernn: An unconditional end-to-end neural audio generation model,” arXiv preprint arXiv:1612.07837, 2016.

[9] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. Courville, and Y. Bengio, “Char2wav: End-to-end speech synthesis,” ICLR, 2017.

[10] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio et al., “Tacotron: Towards end-to-end speech synthesis,” Interspeech, 2017.

[11] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[12] D. Bahdanau, K. H. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” ICLR, 2015.

[13] D. Griffin and J. Lim, “Signal estimation from modified shorttime fourier transform,” IEEE Transactions on acoustics, speech, and signal processing, vol. 32, no. 2, pp. 236–243, 1984.

[14] H. Tachibana, K. Uenoyama, and S. Aihara, “Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention,” ICASSP, 2018.

[15] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, “Attention-based models for speech recognition,” NeurIPS, 2015.

[16] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, “Fastspeech: Fast, robust and controllable text to speech,” NeurIPS, 2019.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez et al., “Attention is all you need,” NeurIPS, 2017.

[18] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, “Neural speech synthesis with transformer network,” AAAI, 2019.

[19] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, “Fastspeech 2: Fast and high-quality end-to-end text to speech,” ICLR, 2021.

[20] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, Deep learning. MIT press, 2016.

[21] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. WardeFarley, S. Ozair et al., “Generative adversarial nets,” NeurIPS, 2014.

[22] D. Rezende and S. Mohamed, “Variational inference with normalizing flows,” ICML, 2015.

[23] A. V. D. Oord, I. Li, I. Babuschkin, K. Simonyan, O. Vinyals, K. Kavukcuoglu et al., “Parallel WaveNet: Fast high-fidelity speech synthesis,” ICML, 2018.

[24] W. Ping, K. Peng, and J. Chen, “ClariNet: Parallel wave generation in end-to-end text-to-speech,” ICLR, 2019.

[25] R. Prenger, R. Valle, and B. Catanzaro, “Waveglow: A flowbased generative network for speech synthesis,” ICASSP, 2019.

[26] D. P. Kingma and P. Dhariwal, “Glow: Generative flow with invertible 1x1 convolutions,” NeurIPS, 2018.

[27] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo et al., “MelGAN: Generative adversarial networks for conditional waveform synthesis,” NeurIPS, 2019.

[28] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz et al., “Highresolution image synthesis and semantic manipulation with conditional gans,” CVPR, 2018.

[29] R. Yamamoto, E. Song, and J.-M. Kim, “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” ICASSP, 2020.

[30] J. Yang, J. Lee, Y. Kim, H.-Y. Cho, and I. Kim, “VocGAN: A High-Fidelity Real-Time Vocoder with a Hierarchically-Nested Adversarial Network,” Interspeech, 2020.
[31] Z. Zhang, Y. Xie, and L. Yang, “Photographic text-to-image synthesis with a hierarchically-nested adversarial network,” CVPR, 2018.
[32] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang et al., “Stackgan++: Realistic image synthesis with stacked generative adversarial networks,” IEEE transactions on pattern analysis and machine intelligence, vol. 41, no. 8, pp. 1947 – 1962, 2018.
[33] J. Kong, J. Kim, and J. Bae, “Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis,” NeurIPS, 2020.
[34] Y. Wang, D. Stanton, Y. Zhang, R.-S. Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, and R. A. Saurous, “Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis,” ICML, 2018.
[35] Y. Lee and T. Kim, “Robust and fine-grained prosody control of end-to-end speech synthesis,” ICASSP, 2019.
[36] T. Raitio, R. Rasipuram, and D. Castellani, “Controllable neural text-to-speech synthesis using intuitive prosodic features,” Interspeech, 2020.
[37] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren et al., “Transfer learning from speaker verification to multispeaker textto-speech synthesis,” NeurIPS, 2018.
[38] Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Z. Chen, R. Skerry-Ryan, Y. Jia, A. Rosenberg, and B. Ramabhadran, “Learning to speak fluently in a foreign language: Multilingual speech synthesis and cross-language voice cloning,” Interspeech, 2019.
[39] J. Kim, S. Kim, J. Kong, and S. Yoon, “Glow-tts: A generative flow for text-to-speech via monotonic alignment search,” NeurIPS, 2020.
[40] Y. Lee, J. Shin, and K. Jung, “Bidirectional variational inference for non-autoregressive text-to-speech,” ICLR, 2020.

약 력

양 진 혁

2018 한동대학교, 컴퓨터공학과 졸업 (학사)
2019 한동대학교, 정보통신공학부 졸업 (석사)
2019~현재 NCSOFT, AI Center Speech AI Lab 음성합성팀 연구원
관심분야: 생성모델, 게임 응용, 서비스 응용
Email: yangyangii@ncsoft.com