The Journal of the Convergence on Culture Technology (JCCT)
Vol. 8, No. 3, pp.469-475, May 31, 2022. pISSN 2384-0358, eISSN 2384-0366

http://dx.doi.org/10.17703/JCCT.2022.8.3.469
JCCT 2022-5-58

d-vector를 이용한 한국어 다화자 TTS 시스템

A Korean Multi-speaker Text-to-Speech System Using d-vector

김광현*, 권철홍**

Kwang Hyeon Kim*, Chul Hong Kwon**

요 약 딥러닝 기반 1인 화자 TTS 시스템의 모델을 학습하기 위해서 수십 시간 분량의 음성 DB와 많은 학습 시간이 요구된다. 이것은 다화자 또는 개인화 TTS 모델을 학습시키기 위해서는 시간과 비용 측면에서 비효율적 방법이다. 음색 복제 방법은 새로운 화자의 TTS 모델을 생성하기 위하여 화자 인코더 모델을 이용하는 방식이다. 학습된 화자 인코더 모델을 통해 학습에 사용되지 않은 새로운 화자의 적은 음성 파일로부터 이 화자의 음색을 대표하는 화자 임베딩 벡터를 만든다. 본 논문에서는 음색 복제 방식을 적용한 다화자 TTS 시스템을 제안한다. 제안한 TTS 시스템은 화자 인코더, synthesizer와 보코더로 구성되어 있는데, 화자 인코더는 화자인식 분야에서 사용하는 d-vector 기법을 적용한다. 학습된 화자 인코더에서 도출한 d-vector를 synthesizer에 입력으로 추가하여 새로운 화자의 음색을 표현한다. MOS와 음색 유사도 청취 방법으로 도출한 실험 결과로부터 제안한 TTS 시스템의 성능이 우수함을 알 수 있다.

주요어 : TTS, 다화자, 화자 임베딩, d-vector

Abstract To train the model of the deep learning-based single-speaker TTS system, a speech DB of tens of hours and a lot of training time are required. This is an inefficient method in terms of time and cost to train multi-speaker or personalized TTS models. The voice cloning method uses a speaker encoder model to make the TTS model of a new speaker. Through the trained speaker encoder model, a speaker embedding vector representing the timbre of the new speaker is created from the small speech data of the new speaker that is not used for training. In this paper, we propose a multi-speaker TTS system to which voice cloning is applied. The proposed TTS system consists of a speaker encoder, synthesizer and vocoder. The speaker encoder applies the d-vector technique used in the speaker recognition field. The timbre of the new speaker is expressed by adding the d-vector derived from the trained speaker encoder as an input to the synthesizer. It can be seen that the performance of the proposed TTS system is excellent from the experimental results derived by the MOS and timbre similarity listening tests.

Key words : TTS, Multi-speaker, Speaker Embedding, d-vector

*정회원, 대전대학교 대학원 정보통신공학과 석사수료 (제1저자)
**정회원, 대전대학교 정보통신·전자공학과 교수 (교신저자)
접수일: 2022년 3월 17일, 수정완료일: 2022년 4월 12일
게재확정일: 2022년 4월 18일

Received: March 17, 2022 / Revised: April 12, 2022
Accepted: April 18, 2022
**Corresponding Author: chkwon@dju.ac.kr
Dept. of Information, Communication, Electronics Engineering, Daejeon Univ, Korea

Ⅰ. 서 론

TTS(Text-to-Speech) 시스템은 입력으로 들어온 텍스트를 여러 개의 모듈로 처리하여 합성음을 생성하여 출력한다. 기존에 상용화 되어 있는 파형 연결 방식의 TTS 시스템[1]은 텍스트 전처리, 구문 및 경계 분석, 발음표기 변환 처리, 운율 분석 및 조절, 음성 DB에서 적합한 단위 음성 선정 및 연결을 통한 합성음 생성 등의 모듈로 구성되어 있다. 딥러닝 기반 End-to-End 방식의 TTS 기술은 파형 연결 방식과 달리 두 개의 모듈로 통합하여 처리한다[2][3]. 이에는 입력 텍스트에서 스펙트로그램을 만드는 synthesizer 모듈과 스펙트로그램에서 합성 파형을 생성하는 보코더가 있다.

딥러닝 기반 1인 화자 TTS 시스템에서 synthesizer와 보코더 모델을 학습하기 위해서는, 전문 성우가 녹음한 수십 시간 분량의 음성 DB와 텍스트 정보가 필요하고, 수 일 또는 수십 일 간의 학습 시간이 요구된다. 이것은 다화자 또는 특정 화자의 개인화 TTS 모델을 학습시키기 위해서는 시간과 비용 측면에서 비효율적 방법이다. 다화자 TTS에서는 여러 명의 전문 성우가 녹음한 음성 DB가 필요하며, 시스템을 구축하기 위해 매우 많은 학습 시간이 요구된다. 개인화 TTS에서는 전문 성우가 아닌 개인이 수십 시간 분량의 음성 DB를 녹음하는 부담이 따른다.

음색 복제(Voice Cloning)는 새로운 화자의 TTS 모델을 생성하기 위하여 화자 인코더 모델을 학습하는 방식을 취한다[4]. 화자 인코더는 수 천 또는 수 만 명의 매우 많은 화자의 음성 DB와 화자 아이디 정보를 사용하여 학습된다. 학습된 화자 인코더 모델을 통해 학습에 사용되지 않은 새로운 화자의 수 초 또는 수 분 분량의 음성 파일로부터 이 화자의 음색을 대표하는 화자 임베딩 벡터를 만들어 해당 화자를 위한 TTS 시스템을 가능하게 한다.

본 논문에서는 음색 복제 방식을 적용한 다화자 TTS 시스템을 제안한다. 본 논문에서 제안한 음색 복제 방식은 [4]에서 제안한 방식에 기반을 둔다. 이 방식은 화자 인코더, synthesizer와 보코더로 구성되어 있는데, 화자 인코더는 화자인식 분야에서 사용되는 d-vector 기법[5]을 적용한다.

본 논문에서는 1장 서론에 이어 2장에서 제안하는 한국어 다화자 TTS 시스템을 기술한다. 3장에서 화자 인코더, synthesizer와 보코더를 학습하는 방법에 대해 설명한다. 4장에서 본 논문에서 제안한 다화자 TTS 시스템의 성능 평가 결과를 제시하며, 5장에서 결론을 맺는다.

Ⅱ. 한국어 다화자 TTS 시스템

본 논문에서는 [4]에서 제안하여 영어에 적용된 음색 복제 기법을 한국어에 적용하여 한국어 다화자 TTS 시스템을 연구한다. 먼저 학습 환경에 맞게 한국어 DB를 가공 처리하고, 화자인증 기술인 GE2E(Generalized End-to-End) 기법[5]을 적용하여 화자 인코더를 구성하고 화자의 특징 벡터인 d-vector를 생성한다. 다음에, 다화자 TTS를 위해 텍스트와 멜-스펙트로그램 데이터에 추가하여 d-vector를 입력으로 주어 Tacotron2[6]를 기반으로 구성된 synthesizer를 학습한다. 그리고 WaveRNN[7] 기반의 보코더로 멜-스펙트로그램을 합성음성으로 변환시켜 원하는 화자의 음성을 생성한다.

본 논문에서 제안한 다화자 TTS 시스템의 구조는 그림 1과 같다[8].

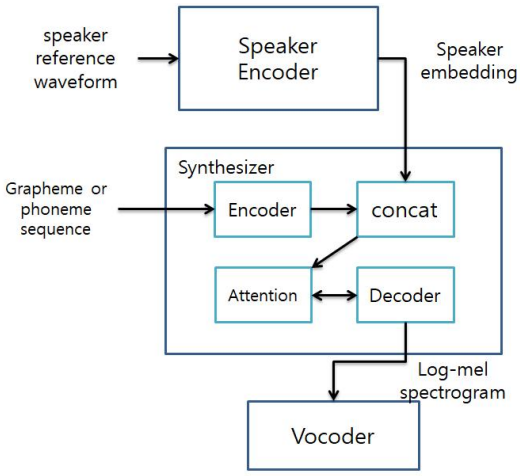

그림 1. 다화자 TTS 시스템 구조
Figure 1. Structure of the multi-speaker TTS system

화자 인코더를 이용하여 임의의 화자의 멜-스펙트로그램을 작고 고정된 크기의 화자 임베딩 벡터인 d-vector로 변환한다. Tacotron2를 기반으로 한 synthesizer의 인코더를 통해 입력 한글 텍스트를 문자 임베딩 벡터로 변환하고, 이 벡터와 d-vector를 연결하여(Concatenate) synthesizer의 어텐션 입력으로 사용한다. Sequence-

to-sequence 방식으로 구성된 synthesis 모듈로 멜-스펙트로그램을 생성하고, WaveRNN 기반으로 구성된 보코더로 멜-스펙트로그램에서 합성음을 생성한다.

III. 모델 학습 방법

이 절에서는 화자 인코더 모델 학습, 시간 정렬 정보 생성과, synthesizer와 보코더 모델을 학습하는 방법에 대하여 설명한다[8]. 다화자 TTS 시스템을 구현하기 위해 Jemine이 제공한 오픈 소스[9][10]를 활용한다.

1. 화자 인코더 모델 학습

화자 인코더 모델을 학습하기 위해서는 화자 아이디 정보가 포함된 매우 많은 화자의 음성 데이터가 필요하다. 그런데 한국어인 경우 이러한 정보가 포함된 공개 음성 DB를 확보하기 어려우므로, 먼저 영어 음성 DB를 이용하여 모델을 학습하고, 학습된 모델을 한국어 음성 DB을 이용하여 추가적으로 학습하는 미세조정(Fine Tuning) 방식을 취한다.

영어 화자 인코더를 학습하기 위해 VoxCeleb1/2[11], LibriSpeech/train-other-550[12]에서 약 8,500명의 화자가 발화한 약 130만개의 음성 파일을 사용한다. 한국어 화자 인코더 학습에 사용된 음성 DB의 화자 수는 435명이며 문장 수는 87,403개이다. 이 DB에는 네이버 클로바에서 제공하는 ClovaCall[13], OpenSLR에서 제공하는 Zeroth-Korean[14]과, 한국전자통신연구원에서 제공하는 음성 학습데이터[15] 등이 포함되어 있다.

화자 인코더 모델을 학습하는 과정은 다음과 같다. 음성 파일에서 화자 인코더 모델에서 사용하는 특징 파라미터인 40 차원의 멜-스펙트로그램을 추출한다. 이 파라미터를 이용하여 화자 인코더 모델을 학습하면 256 차원의 d-vector를 생성할 수 있다.

2. Synthesizer 학습을 위한 시간 정렬 정보 생성

본 논문에서 제안한 다화자 TTS 시스템의 synthesizer는 Tacotron2를 기반으로 한다. Tacotron2의 핵심은 어텐션 모듈로서, 이것은 텍스트와 멜-스펙트로그램이 시간적으로 정렬(Alignment)되게 만든다. 즉, 텍스트의 특정 문자가 멜-스펙트로그램의 어떤 부분에 해당하는지 시간적으로 정렬한다. 정렬 학습이 제대로 수행되지 않으면 좋은 합성음을 생성할 수 없다. 기존의 Tacotron2에서는 텍스트와 멜-스펙트로그램을 입력으로 받아들여 학습을 반복적으로 수행하면서 이 둘 사이의 시간 정렬을 맞춘다.

본 논문에서는 정렬을 올바르게 수행하도록 텍스트와 멜-스펙트로그램 데이터에 추가로 시간 정보를 입력으로 제공한다. 이를 위해 음성인식에서 널리 사용되는 강제 정렬(Forced Alignment) 방식을 통해 시간 정보가 포함된 TextGrid 파일을 생성한다. 시간 정보는 음성 파일의 전체 음성 길이, 각 음소와 어절의 시작 시각과 끝나는 시각이다.

한국어 자동강제정렬(Korean Forced Aligner) 툴[16]과 Kaldi 툴[17]을 사용하여 TextGrid 파일을 생성한다. 자동강제정렬 툴은 음성 파일과 해당 음성 파일을 전사한 텍스트 파일을 입력으로 하여 문장을 음소나 어절 단위로 나누어 시간 정보를 출력으로 생성한다.

3. Synthesizer와 보코더 모델 학습

Synthesizer와 보코더는 한국어 음성 DB만 이용하여 학습하고, 이 두 개 모듈의 음성 DB는 동일하다. 학습에 사용된 음성 DB의 전체 화자 수는 1,060명이고 문장 수는 359,284개이다. 이 DB는 화자 인코더에 사용된 음성 데이터를 포함하고 있고, 추가로 한국지능정보사회진흥원에서 제공하는 한국어 자유 발화 음성 데이터[18]가 포함되어 있다. 이 DB의 화자 수는 625명이고 문장 수는 271,881개이다.

Synthesizer와 보코더 모델을 학습하기 위하여, 음성 파일에서 프레임 크기를 50msec로 하여 12.5msec 간격마다 크기가 800인 FFT(Fast Fourier Transform)를 추출하여 80 차원의 멜-스펙트로그램을 구한다. Synthesizer는 멜-스펙트로그램과 이에 해당하는 텍스트 파일, 그리고 화자 인코더 모델을 통해 도출된 화자 임베딩 벡터인 d-vetcor를 입력으로 하여 학습한다. 화자 인코더 모델에 멜-스펙트로그램을 입력으로 주어 해당 음성 파일의 d-vector을 생성하고 이를 synthesizer에 사용한다. 보코더는 멜-스펙트로그램과 해당 음성 파일을 입력으로 하여 학습한다. 여기에 사용된 멜-스펙트로그램은 synthesizer에서 사용한 것과 동일하다.

IV. 모델 학습 결과

1. 실험 환경

화자 인코더, synthesizer와 보코더 모델을 학습하기 위한 시뮬레이션 하드웨어 환경은, 운영체제 Ubuntu 18.04LTS, CPU 인텔 i9-10900K, GPU Nvidia GTX 3090 2개, 메인 메모리 64GB 등이다. 필요한 소프트웨어는 Python 버전 3.7, PyTorch 버전 1.8.1, CUDA 버전 11.1, Kaldi 버전 5.3 등이다.

2. 화자 인코더 학습 결과

화자 인코더의 학습 결과는 그림 2, 3과 같은 그래프로 확인할 수 있다. 이 그래프는 다차원을 2차원으로 보여준 그래프로 가로축과 세로축의 의미는 없다. 학습이 잘못되거나 학습 시간이 적으면 그림 2와 같이 다른 색의 점들이 섞여 있음을 볼 수 있다. 화자 인코더 학습이 진행되면서 그림 3과 같이 같은 색 점끼리 뭉치게 되는데 이로부터 학습이 잘 진행되었음을 알 수 있다.

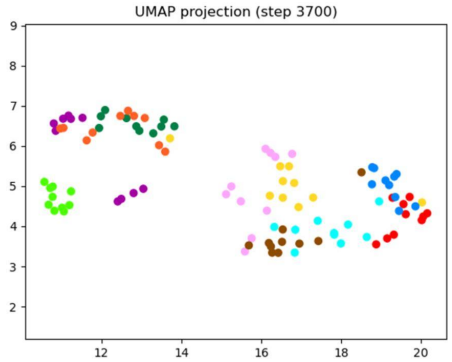

그림 2. 화자 인코더 학습이 잘못된 경우의 그래프
Figure 2. Graph of the case of wrong trained speaker encoder

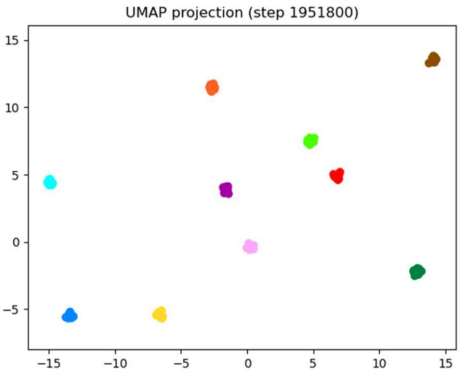

그림 3. 화자 인코더 학습이 잘된 경우의 그래프
Figure 3. Graph of the case of well trained speaker encoder

3. Synthesizer 학습 결과

Synthesizer의 학습 결과는 어텐션 그래프로 확인할 수 있다. 이 그래프에서 가로축은 synthesizer의 디코더 스텝(출력 데이터인 멜-스펙트로그램)을, 세로축은 인코더 스텝(입력 데이터인 텍스트)을 나타내며, 학습 데이터의 정렬을 보여 준다. 학습이 잘 되면 그래프에 가늘고 선명한 대각선이 보인다. 학습이 잘못되면 정렬이 잘되지 않아 그림 4와 같이 선 모양이 아닌 퍼져 있는 패턴이 보인다.

그림 5와 6은 학습 시간이 같은 조건에서, Ⅲ.절에서 기술한 시간 정렬 정보의 사용 유무에 따른 결과를 보여준다. 그림 5는 이 정보를 사용하지 않은 경우로 선이 선명하지 않고 번져 있는 것을 볼 수 있다. 그림 6은 시간 정렬 정보를 사용한 경우로 선이 가늘고 선명하게 보이므로, 이 정보를 추가하여 학습하는 것이 synthesizer의 성능을 향상시킬 수 있음을 알 수 있다.

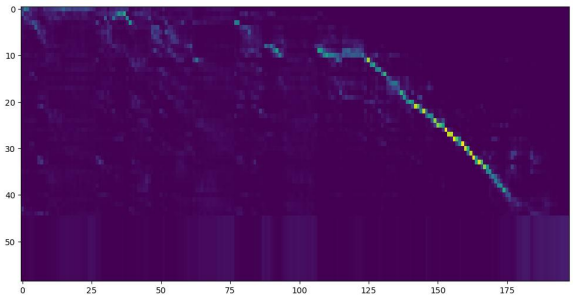

그림 4. Synthesizer 학습이 잘못된 경우의 어텐션 그래프
Figure 4. Attention graph of the case of wrong trained synthesizer

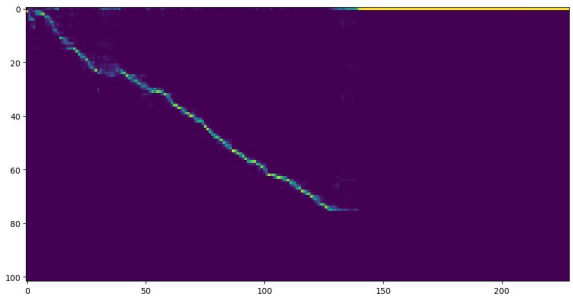

그림 5. 시간 정렬 정보를 사용하지 않아 synthesizer 학습이 잘못된 경우의 어텐션 그래프
Figure 5. Attention graph of the case of wrong trained synthesizer without time alignment information

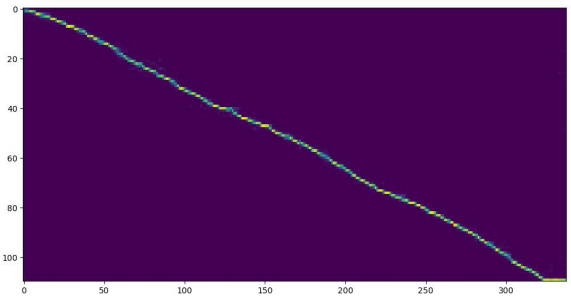

그림 6. 시간 정렬 정보를 사용하여 synthesizer 학습이 잘된 경우의 어텐션 그래프
Figure 6. Attention graph of the case of well trained synthesizer with time alignment information

V. TTS 합성음 성능 평가

성능 평가에 사용되는 합성음 샘플을 생성하기 위해 필요한 입력 데이터는 합성을 원하는 텍스트 데이터와 수 초 길이의 음성 파일이다. 이 음성 파일로부터 화자 인코더를 통해 이 화자의 음색을 표현하는 d-vector를 생성하여 합성한다.

1. MOS(Mean Opinion Score) 평가

제안한 TTS 시스템의 성능 평가를 위해 주관적 음질 평가 지표로 널리 사용되는 MOS를 사용한다. MOS 방식은 합성음의 품질을 자연성과 명료도 측면에서 평가하는 주관적 청취 평가 방법이다. 제안한 TTS 시스템에서 합성된 음성 샘플을 평가자 그룹에게 들려주고, 합성음의 가장 낮은 품질에 1점을, 가장 높은 품질에 5점을 부여하여, 평가 대상자들이 합성음의 품질에 대하여 1점에서 5점까지 점수를 매긴다. 이러한 방식으로 합성음의 자연성과 명료도에 대해 개별 문장마다 점수를 부여한 다음, 그 점수들의 평균값을 구하여 합성음의 수준을 수치화 한다.

표 1. MOS 평가 결과
Table 1. Evaluation results for MOS

합성음	1	2	3	4	5	6
MOS	4.23	4.35	4.11	4.05	4.29	3.88
합성음	7	8	9	10	11	12
MOS	3.76	4.16	3.52	4.11	3.94	3.88

표 1에 합성음 샘플 각각에 대한 피험자들의 평균 점수가 보인다. MOS 평가에 참여한 피험자는 정상 청력을 가진 20-30대 16명이고, 합성음성 샘플은 12개이다. 전체 MOS 평균값은 4.02로, 일반적으로 MOS 4.0 이상은 매우 우수한 것으로 평가한다.

2. 음색 유사도 청취 평가

음색 유사도 청취 평가 방식은 목표화자의 합성음과 원음성과의 음색 유사도를 평가하는 주관적 평가 방법이다. 음성 합성기로부터 생성된 합성음과 원음성의 쌍을 평가자 그룹에게 들려주고, 이 두 음성이 얼마나 유사한지를 선택하는 방식으로 진행된다. 평가를 진행할 때 합성음성의 자연스러움과 발음의 명료도를 고려하지 않고 단지 음색의 유사성만을 평가한다.

표 2에 목표화자 각각에 대한 유사도 평가 결과가 보인다. 목표화자의 원음성과 합성음성을 한 세트로 하여 목표화자 12명의 합성음 데이터로 평가를 진행하며, 평가 피험자는 MOS 평가자와 동일하게 16명이다. 표의 세로는 목표화자 12명을 나타내고, 가로의 1~4는 각각 '확실하게 같다', '확실하지 않지만 같다', '확실하지 않지만 다르다', '확실하게 다르다'를 의미한다. 청취 평가 결과 '확실하게 같다'는 29.17%, '확실하지 않지만 같다'는 52.60%, '확실하지 않지만 다르다'는 17.71%, '확실하게 다르다'는 0.52%로, 합성음이 목표 화자의 원음성과 유사하다고 평가됨을 알 수 있다.

표 2. 유사도 평가 결과
Table 2. Evaluation results for voice similarity

평가척도 / 목표화자	1	2	3	4
1	6	7	3	0
2	4	9	2	1
3	2	11	3	0
4	4	8	4	0
5	4	11	1	0
6	7	5	4	0
7	3	10	3	0
8	8	6	2	0
9	5	7	4	0
10	6	8	2	0
11	3	8	5	0
12	4	11	1	0
합계	56	101	34	1
비율(%)	29.17	52.60	17.71	0.52

Ⅵ. 결 론

딥러닝 기반 TTS 시스템의 모델을 학습하기 위해서는 화자마다 수십 시간 분량의 음성 DB와 매우 많은 학습 시간이 필요하다. 다화자 TTS 시스템에서는 다수의 화자가 녹음한 음성 DB가 필요하며, 매우 많은 시스템 구축 시간이 요구된다. 본 논문에서는 학습에 사용되지 않은 화자의 적은 음성 데이터를 이용하여 짧은 학습 시간에 이 화자의 음색을 표현하는 다화자 음성합성 시스템을 제안한다.

본 논문에서는 영어에 적용된 음색 복제 기법을 한국어에 적용하여 한국어 다화자 TTS 시스템을 연구한다. 이 시스템은 화자 인코더, synthesizer와 보코더로 구성되어 있다. 화자 인코더는 GE2E 기법을 적용하여 구성하고 화자의 음색을 대표하는 d-vector를 생성한다. Tacotron2를 기반으로 synthesizer를 구성하는데, 텍스트, 멜-스펙트로그램과 d-vector를 입력으로 하여 학습한다. 보코더는 WaveRNN 기반으로 구성하여, 멜-스펙트로그램에서 합성음성을 생성한다.

화자 인코더 학습 결과는 이차원 그래프를 통해 같은 색 점끼리 뭉치는 패턴으로부터 학습이 잘 진행되었음을 확인하였다. Synthesizer의 학습 결과는 어텐션 그래프로 확인하였는데, 학습이 잘 된 경우 선이 가늘고 선명하게 보이는 패턴을 보였다. TTS 합성음의 성능은 주관적 음질 평가 방식인 MOS와 음색 유사도 청취 방식으로 평가하였고 우수한 결과를 보여주었다.

향후에, 화자의 음색을 대표하는 방식으로 d-vector 이외에 화자인식 분야에서 성능이 우수한 방식을 연구하여, 성능이 보다 향상된 한국어 다화자 TTS 시스템에 대해 연구를 수행할 계획이다.

References

[1] A. J. Hunt and A. W. Black, “Unit selection in a concatenative speech synthesis system using a large speech database”, Proceedings of the IEEE International Conference on Acoustics, Speech, Signal Processing (ICASSP) 1996, pp. 373-376, 1996, DOI: 10.1109/ICASSP.1996.541110

[2] C. H. Kwon, “Performance comparison of state-of-the-art vocoder technology based on deep learning in a Korean TTS system”, The Journal of the Convergence on Culture Technology (JCCT), Vol. 6, No. 2, pp. 509-514, 2020, DOI: 10.17703/JCCT.2020.6.2.509

[3] M. S. Jo and C. H. Kwon, “A multi-speaker speech synthesis system using x-vector”, The Journal of the Convergence on Culture Technology (JCCT), Vol. 7, No. 4, pp. 675-681, 2021, DOI:10.17703/JCCT.2021.7.4.675

[4] Y. Jia, Y. Zhang, R. Weiss, et al., “Transfer learning from speaker verification to multispeaker text-to-speech synthesis”, ArXiv:https://arxiv.org/pdf/1806.04558.pdf, Jan. 2019

[5] A. Papir, I. Wan, Q. Wang, et al., “Generalized end-to-end loss for speaker verification”, ArXiv: https://arxiv.org/pdf/1710.10467.pdf, Nov. 2020

[6] J. Shen, R. Pang, R. J. Weiss, et al., “Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions”, Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2018, pp. 4779-4783, 2018, DOI: 10.1109/ICASSP.2018.8461368

[7] E. Elsen, N. Kalchbrenner, K. Simonyan, et al,, “Efficient neural audio synthesis”, ArXiv. https://arxiv.org/pdf/1802.08435.pdf, June 2018.

[8] K. H. Kim, “A study on multi-speaker TTS using speaker recognition technology”, *Master Thesis*, Graduate School of Daejeon Univ. 2022

[9] C. Jemine, “Real-time voice cloning”, Master Thesis, Liege University, 2019

[10]Real-time Voice Cloning, https://github.com/CorentinJ/Real-Time-Voice-Cloning

[11]A. Nagrani, J. S. Chung, A. Zisserman, “VoxCeleb: A large-scale speaker identification dataset”, Proceedings of the Interspeech 2017, pp. 2616-2620, 2017, DOI:10.1109/ICASSP.2018.8461375

[12]H. Zen, V. Dang, R. Clark, et al., “LibriTTS: A corpus derived from LibriSpeech for text-to-speech”, Proceedings of the Interspeech 2019, pp. 1526-1530, 2019, DOI:10.21437/Interspeech.2019-2441

[13]J. W. Ha, K. H. Nam, J. Kang, et al., “ClovaCall: Korean goal-oriented dialog speech corpus for automatic speech recognition of contact centers”, Proceedings of the Interspeech 2020, pp. 409-413, 2020, DOI:10.21437/Interspeech.2020-1136

[14]Zeroth-Korean, Korean open source speech corpus for speech recognition by Zeroth project, https://www.openslr.org/40/

[15]한국전자통신연구원, 음성 학습 데이터, https://aiopen.etri.re.kr/service_dataset.php?category=voice

[16]Korean Forced Aligner, https://github.com/hyung8758/Korean_FA
[17]D. Povey, A. Ghoshal, G. Boulianne, et al., "The Kaldi speech recognition toolkit", Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU) 2011, 2011
[18]한국지능정보사회진흥원, AI Hub, 한국어 자유 발화 음성 데이터, https://aihub.or.kr/aidata/105

※ 이 논문은 한국연구재단 지역대학 우수과학자 지원 사업(NRF-2020R1I1A3052136)에 의해 연구되었음