음성 명령어의 음절 분석을 위한 딥러닝 신경망 설계

민경진*, 백지엽*, 조성동*, 이강휘*, 김정환*
㈜피앤씨솔루션*

Design Deep Neural Network for syllable analysis of voice commands

Kyoung-Jin Min*, Jee-Yeop Back*, Sung-Dong Cho*, Kang-Hwi Lee*, Jeong-Hwan Kim*
P&C Solution Co., Ltd.*

Abstract - 메타버스의 시대가 도래하면서 Virtual Reality (VR), Augmented Reality(AR) 산업이 큰 각광을 받고 있다. 일반적으로 VR 장치에는 특수한 컨트롤러가 사용되지만 AR 장치는 현실과 접목되어야 하기에 추가적인 컨트롤러를 사용할 수 없다. 따라서 AR 기기의 제어는 휴대폰에 연결하여 컨트롤러를 대신하거나, 카메라, 마이크와 같은 센서로 신체의 정보를 활용해야 한다. 본 논문은 AR 기기를 음성 명령어로 제어하기 위해 음성을 음절 단위로 분석하는 딥러닝 모델을 제안한다.

1. 서 론

메타버스의 시대가 도래하면서 가상현실(Virtual Reality;VR), 증강현실(Augmented Reality;AR) 산업이 큰 각광을 받고 있다. 이 중에서도 AR은 사용자의 초점을 방해하지 않으면서 정보를 표시하기 때문에 현실의 작업과 관련 정보 사이의 격차를 해소함으로써 인지 부하를 줄이는 매체이다[1]. AR은 현실과 접목된 분야인 만큼 편의성을 띠고 있기 때문에, 대부분의 VR 기기에서 사용되는 컨트롤러 같은 장치[2]를 포함하고 있지 않다. 따라서 AR 기기의 제어는 휴대폰에 연결하여 컨트롤러를 대신하거나, 사람의 신체 정보를 이용해야 한다.

본 논문은 음성으로 AR 기기를 제어하기 위한 기술 연구이며, 딥러닝 모델을 활용하여 음성을 음절 단위로 해석하고자 한다.

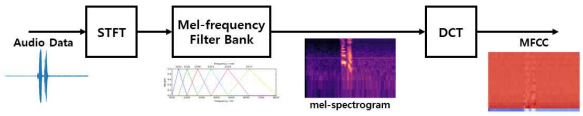
2. 본 론

2.1 음성 신호의 특징 추출

음성은 연속적인 시간에 의존하는 음절의 조합으로 이루어져 있으므로 시계열 분석으로 특징을 추출하고 분석한다. 일반적으로 실제 사람이 소리를 인식하는 특징을 반영한 MFCC(Mel Frequency Cepstral Coefficient) 알고리즘[3]을 주로 사용하며, MFCC 알고리즘의 과정은 다음과 같이 진행된다.

1. 음성 신호를 일정 구간 프레임으로 나누어 주파수를 분석하는 Sort-Time Fourier Transform(STFT)를 적용하여 일정한 시간 단위의 주파수 특징인 스펙트로그램(spectrogram)을 구한다.
2. 주파수 대역의 에너지를 분석하기 위한 Mel-frequency Filter Bank를 스펙트로그램에 적용하여 Mel-spectrogram을 구한다.
3. Mel-spectrogram 결과를 다시 Discrete Fourier Transform(DCT)을 적용한다.
4. DCT 결과는 Log 연산으로 결과 크기의 범위를 조절한다.

음성 신호의 MFCC를 적용한 전체 과정과 결과는 그림 1에서 보여준다.



〈그림 1〉 MFCC 추출 과정

2.2 딥러닝의 시계열 해석

음성의 MFCC 특징값 벡터는 시간의 정보를 나타내는 x축과 주파수 에너지를 표현하는 y축의 2차원 이미지로 표현된다. 이 시계열 정보는 딥러닝에서 주로 사용하는 CNN(Convolutional Neural Network)의 커널 stride 연산 개념과 일치한다. 필터 커널은 x축을 따라 이동하면서 연산하는 영상 컨볼루션 처리이며 stride x축으로 이동하는 단위를 말한다. 시계열 해석의 딥러닝 모델은 순환신경망(RNN)을 활용하지만, 본 논문에서는 CNN의 구조로 해석하는 방법으로 해석하고자 한다.

2.3 실험 내용

2.3.1 음성 명령어 수집 및 라벨링

음성 데이터는 48,000Hz 샘플레이트로 앞뒤 1초의 공백을 두고 2초간 발화하도록 제한하여 스마트폰에서 수집되었다. 음성 데이터는 남성 50명, 여성 50명을 대상으로 31개 타겟 명령어는 5회, 명령어가 아닌 단어 30개씩 1회로 녹음하였다. 표 1은 실험에서 사용된 음성 명령어의 목록이며, None은 명령어가 아닌 다양한 단어를 의미한다.

본 논문에서는 음성을 8구간의 음절 단위로 나누었으며, 명령어의 음절 54개와 침묵(-), None을 포함한 56개를 구분하였다. 예를 들어 '선택'의 경우 ['-', '-', '선', '택', '-', '-', '-', '-']과 같이 라벨링이 구성된다.

〈표 1〉 음성 명령어

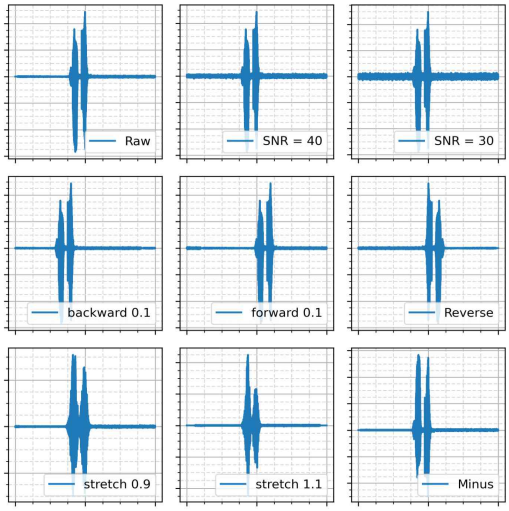
0	None	8	음성명령어	16	재생	24	전체화면
1	선택	9	촬영	17	되감기	25	이동
2	클릭	10	녹화	18	빨리감기	26	멈춤
3	닫기	11	정지	19	처음	27	모든창보기
4	홈	12	아래로	20	소리작게	28	전화
5	종료	13	위로	21	소리크게	29	통화
6	어둡게	14	다음	22	화면크게	30	수락
7	밝게	15	이전	23	화면작게	31	거절

2.3.2 데이터 증강

부족한 데이터에서 모델의 성능을 높이기 위한 방법으로 데이터를 증강시킨다. 음성 데이터도 증강이 필요하기에 표 2와 같은 방법으로 잡음추가, 시간 변화, 구간 이동, $x \cdot y$축 반전을 수행하였고, 그림 3에서 원본 음성 신호를 증강한 결과 샘플을 보여주고 있다.

〈표 2〉 데이터의 증강 방법

구분	내용	데이터 설정
백색잡음	기존 음성 신호와 백색잡음의 비율을 지정한 SNR에 맞추어 합성한다.	SNR 40 dB
		SNR 30 dB
배속조절	기존 음성 신호의 배속을 조절하여 음성이 늘어지거나, 빨라지도록 한다.	0.9배속
		1.1배속
위치이동	음성을 설정한 비율만큼 앞뒤로 이동한다.	forward 10%
		backward 10%
신호반전	음성 신호의 부호를 바꾼다.	Minus
시간반전	음성의 시간을 거꾸로 뒤집는다.	Reverse

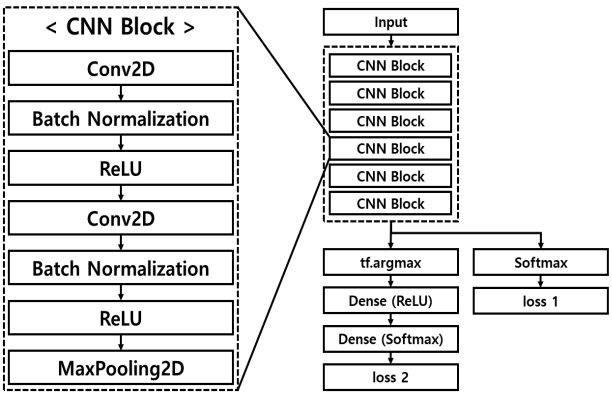


〈그림 3〉 딥러닝 모델의 학습에 사용되는 증강된 음성 데이터

2.3.3 모델 설계

MFCC의 파라미터인 frame length는 1024, frame step은 256으로 Windowing과 FFT를 설정하면 특징벡터는 247×247의 맵이 생성되고 이를 딥러닝의 입력 영상으로 사용한다.

딥러닝의 모델 구성은 CNN, 배치 정규화, maxpooling 층으로 조합된 블록을 구성해 6층을 쌓고 마지막 부분을 Fully Connect 층으로 구성한다. 모델을 학습하는 과정에서 최종적인 출력인 32개의 단어 명령어 분류를 통해 업데이트해주면서 중간층에서 해석되는 음절 단위의 분석이 CNN 블록의 가중치를 추가적으로 업데이트 해준다. 그림 4는 CNN 블록과 전체 구조를 보여준다.



〈그림 4〉 모델의 구조

2.4 실험 결과

화자와 명령어에 따라 랜덤하게 8:2의 비율로 학습(691,011개)과 테스트(172,770개) 데이터를 구성하였고 표 3은 딥러닝 모델이 각 명령어에 대한 분류를 했을 때의 정확도를 보여준다.

〈표 3〉 각 명령어에 대한 정확도

0	None	2.07 %	16	재생	100.00 %
1	선택	100.00 %	17	되감기	100.00 %
2	클릭	100.00 %	18	빨리감기	100.00 %
3	닫기	100.00 %	19	처음	100.00 %
4	홈	100.00 %	20	소리작게	100.00 %
5	종료	100.00 %	21	소리크게	100.00 %
6	어둡게	100.00 %	22	화면크게	100.00 %
7	밝게	100.00 %	23	화면작게	100.00 %
8	음성명령어	100.00 %	24	전체화면	100.00 %
9	촬영	100.00 %	25	이동	100.00 %
10	녹화	100.00 %	26	멈춤	100.00 %
11	정지	100.00 %	27	모든창보기	100.00 %
12	아래로	100.00 %	28	전화	100.00 %
13	위로	100.00 %	29	통화	99.98 %
14	다음	100.00 %	30	수락	100.00 %
15	이전	100.00 %	31	거절	100.00 %

None 라벨은 5400개 중 112개만을 분류하였다. None에 사용된 데이터는 학습에 사용되지 않은 음성 명령어가 음절이 비슷한 타겟 명령어로 분류되고 있다. 하지만, 대부분의 명령어는 100%로 분류한 것을 확인할 수 있고, '통화'의 경우 5400개 중 한 개를 제외한 나머지를 분류하였다. None 라벨을 제외하고 명령어만을 대상으로 한 정확도는 99.9994%로, '통화'에서 한 개의 오답을 제외하고 모든 명령어를 분류하였다.

3. 결 론

본 논문은 음성 명령어 인식을 위한 음성 데이터의 음절 단위 분석하는 방법을 제안하였다. 딥러닝의 CNN의 Stride 방식은 시계열의 특성과 일치하며 이는 4초 음성을 8개의 음절 단위로 해석 하였다. 타겟 명령어에 대해서는 99% 이상의 높은 정확도를 보여주지만 학습에 사용되지 않는 명령어인 None 라벨은 약 2%의 낮은 정확도를 보인다. 이를 자연어처리 분야에서 OOV (Out-of-Vocabulary)라고 하며, OOV 문제를 줄이기 위한 연구[4, 5]가 진행되고 있다. 본 연구에서도 음성을 음절 단위로 분리하여서는 OOV 문제를 해결하지 못하였으나 앞으로도 이를 해결하기 위한 세밀한 딥러닝의 모델 연구를 진행해야 할 것으로 사료된다.

[참 고 문 헌]

[1] Chandan K. Sahu, Crystal Young, Rahul Rai, "Artificial intelligence (AI) in augmented reality (AR)-assisted manufacturing applications: a review", International Journal of Production Research, Vol. 59, No. 16, pp. 4903-4959, 2021.

[2] 이하섭, "VR, AR 최신 기술 동향", 정보통신기획평가원 주간기술동향, Vol.1965, pp. 2-12, 2020.

[3] Wei HAN, Cheong-Fat Chan, Chiu-Sing Choy, Kong-Pang Pun, "An Efficient MFCC Extraction Method in Speech Recognition", IEEE International Symposium on Circuits and Systems (ISCAS), pp. 145-148, 2006.

[4] B. Logan, J.-M. Van Thong, P.J. Moreno, "Approaches to reduce the effects of OOV queries on indexed spoken audio", IEEE Transactions on Multimedia, Vol. 7, No. 5, pp. 899-906, 2005.

[5] Savitha Murthy, Dinkar Sitaram, Sunayana Sitaram, "Effect on TTS Generated Audio on OOV Detection and Word Error Rate in ASR for Low-resource Languages", Interspeech, pp. 1026-1060, 2018.