

## 딥러닝을 이용한 청각장애인 위험방지 및 STT 서비스의 통합 플랫폼 구현

정 예 진<sup>1</sup> · 정 윤 아<sup>1</sup> · 김 세 하<sup>1</sup> · 김 동 회<sup>2\*</sup>

<sup>1</sup>강원대학교 IT대학 전기전자공학과 학사과정

<sup>2\*</sup>강원대학교 IT대학 전기전자공학과 교수

# Implementation of Integrated Platform of Risk Prevention and STT Service for the Deaf Using Deep Learning

Yae-Jin Jeong<sup>1</sup> · Yun-A Jung<sup>1</sup> · Sae-Ha Kim<sup>1</sup> · Dong-Hoi Kim<sup>2\*</sup>

<sup>1</sup>Undergraduate, Department of Electrical and Engineering, Kangwon National University, Chuncheon, Korea

<sup>2\*</sup>Professor, Department of Electrical and Engineering, Kangwon National University, Chuncheon, Korea

### [요 약]

청각장애인은 사이렌 소리가 발생하거나 자동차 경적소리가 울려도 해당 소리의 발생을 감지하기 어렵다. 본 논문에서는 일상 생활 속 발생하는 위험 소리를 학습시킨 FCN 모델을 기반으로 청각장애인을 위한 통합 웹 플랫폼을 제시한다. 첫째, 제안하는 플랫폼을 통해 사용자의 주변에서 특정 소리가 발생했음을 파악한다면, 이전보다 사고에 신속하게 대처할 수 있다. 둘째, 플랫폼에서 제공되는 음성-텍스트 변환 서비스를 바탕으로 수화를 알지 못하는 상대방과 원활한 의사소통이 가능하다. 마지막으로, 웹 크롤링을 통해 청각장애인을 위한 최신 이슈와 구직 정보 등 뉴스를 알려주는 서비스로 청각장애인이 무분별한 뉴스를 보기보다 필요한 정보를 추출하여 제공하므로 편리하게 정보를 얻을 수 있는 플랫폼을 제안한다. 실험 결과, MFCC를 추출하여 딥러닝으로 구현한 소리분류는 약 91.07%의 정확도를 보였다.

### [Abstract]

It is difficult for hearing impaired people to detect the occurrence of the sound even if a siren sound or a car horn is sounded. In this paper, we present an integrated web platform for the deaf based on the FCN model that learns dangerous sounds occurring in everyday life. First, if it is determined that a specific sound has occurred around the user through the proposed platform, it is possible to cope with the accident more quickly than before. Second, based on Speech-To-Text services provided on the platform, it is possible to communicate smoothly with others who do not know sign language. Finally, through web crawling, it is a service that informs news such as the latest issues for the deaf and job search information, and it presents a platform that allows hearing-impaired people to conveniently obtain information because it extracts and provides necessary information rather than watching reckless news. The experimental result of sound classification which was implemented as deep learning by extracting MFCC, showed an accuracy of about 91.07%.

**색인어** : 인공지능, 딥러닝, 합성곱 신경망, 파이썬, 소리 분류, 웹 크롤링, 웹 프레임워크, 음성-텍스트 변환

**Keyword** : Artificial Intelligence, Deep Learning, Convolution Neural Network, Python, Sound classification, Web Crawling, Web Framework, Speech-To-Text

<http://dx.doi.org/10.9728/dcs.2022.23.8.1459>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Received** 18 July 2022; **Revised** 04 August 2022

**Accepted** 08 August 2022

**\*Corresponding Author; Dong-Hoi Kim**

**Tel:** +82-33-250-6349

**E-mail:** donghk@kangwon.ac.kr

## I. 서론

취약 계층 중 청각장애인은 각종 안전사고의 위험이나 들어야 하는 소리를 듣지 못하는 상황에 놓여있다. 화재 등 재난 상황이나 위험하다는 상황을 알리는 사이렌 소리는 물론, 차의 경적 소리조차 인지하지 못하기 때문에 사고를 인식할 수 없다. 사람들이 한꺼번에 대피하는 긴급 상황에도 청각장애인들은 두려움 속에 우왕좌왕할 수 밖에 없다. 실내에서는 강아지의 짖는 소리나 아기 울음소리를 듣지 못하여 빠른 대처를 하지 못하면, 큰 사고로 이어질 수 있다. 본 논문에서 구현한 플랫폼은 이러한 문제점들을 개선하기 위해 위험한 상황에서 발생하는 소리를 인식 및 분류할 수 있는 웹페이지 및 어플리케이션을 구축하고자 한다. 딥러닝으로 특정 소리들을 학습하고 사용자 주변에서 해당 소리가 인식되면 해당 소리가 발생했음을 알림으로써 안전사고를 예방할 수 있다.

청각장애인들은 비장애인과 소통할 때, 수화를 모르는 상대방과 의견을 주고받기에는 한계가 있다. 특히 긴급하거나 위급한 상황에 놓였을 때, 수화에 대해 잘 알지 못하는 비장애인에게 자신의 의견을 전달하고 상대방의 의사를 파악하기에는 어려움이 있다. 이에 대한 사례로, 청각장애인 B씨는 팔을 다쳐 병원에 갔지만 수어 통역사를 부르려 했지만 코로나19로 통역 서비스를 제공받을 수 없어서 진료료 거절당했다. 2021년 8월 기준, 수어통역사가 상주하는 의료기관은 2개소에 불과하다. 또한, 2019년 뉴욕에서는 청각장애인이 사고 현장에서 경찰과의 소통이 되지 않아서 범죄를 저지르지 않았음에도 불구하고 교도소로 이송되는 사건이 발생했다. 본 논문에서 구현한 플랫폼의 음성-텍스트 변환(STT; Speech-to-Text)을 통해 장애인과 비장애인 사이의 원활한 의사소통이 이루어질 수 있도록 하였다. 또한, 통화가 불가피한 상황에서, 해당 기능을 이용하면 상대방의 목소리를 문자로 출력하는 것이 가능하다.

특정 소리를 인식하여 사용자에게 소리의 발생을 알려주거나 음성을 텍스트로 변환하는 서비스는 이미 존재하고 있다. 하지만 본 논문에서는 특정 소리의 발생을 알리는 기능과 더불어, 수화를 알지 못하는 상대방과 원활한 의사소통이 가능하도록 음성-텍스트 변환 서비스를 제공하고 청각장애인 관련 뉴스들을 한눈에 확인할 수 있는 페이지를 제작하여 청각장애인을 위한 통합 플랫폼을 구현하고자 하였다. 해당 플랫폼은 PC와 모바일 버전으로 구동이 가능하여 사용자가 선호하는 버전으로 실행할 수 있다.

본 논문의 II장의 관련 연구에서는 딥러닝 기술을 바탕으로 한 소리 분류 기술을 설명하며, III장에서는 제안하는 청각장애인을 위한 소리 분류 및 음성-텍스트 변환 서비스를 제공하는 웹 어플리케이션의 구성을 설명하며, IV장에서는 실험에서 사용된 소리 분류를 위한 딥러닝 모델을 소개한다. V장에는 음성-텍스트 변환 서비스를 설명하였고, VI장에서는 웹 크롤링 기술을 바탕으로 한 서비스를 설명한다. 이를 바탕으로 마지막 VII장에서 진행한 프로젝트 결과를 정리한다.

## II. 관련 연구

### 2-1 딥러닝

딥러닝(Deep Learning)은 인공지능의 한 분야로, 깊은 인공신경망 즉, 연속된 많은 층의 인공신경망을 사용하여 학습을 수행하는 머신러닝 기법이다. 실제 인간의 뇌는 매우 깊은 구조로 연결된 뉴런들로 이루어져있다는 점에서 딥러닝은 진보된 인공지능 기술이라고 할 수 있다. 기존의 머신러닝에서는 학습 데이터로부터 특징을 추출하는 과정에 인간이 관여했었지만, 딥러닝이 등장하면서 특징이 자동으로 추출되는 것이 가능해졌다[1].

딥러닝은 현재 음향 인식, 이미지 분류, 객체 탐지, 자연어 처리 등 다양한 분야에 활용되고 있다. 딥러닝이 적용된 신경망으로는 Convolutional Neural Network(CNN), Recurrent Neural Network(RNN), Fully Connected Network(FCN) 등이 있다. 오디오와 같은 시계열 데이터에 대한 딥러닝에는 RNN과 CNN이 많이 활용되고 있다[2]. CNN은 사물을 인식하는 데 뛰어난 성능을 보이는 신경망으로, 사진 자동 태깅 등 이미지 관련 서비스에 적극 활용되고 있다. CNN에 의해 도출된 이미지의 벡터값을 RNN에 입력하여 이미지를 설명하는 문장을 생성하는 기술도 공개되었다[3]. FCN 기반의 신경망을 활용하여 산불 피해를 탐지하는 기법이 개발된 사례도 있다[4].

CNN은 데이터 분류를 위한 인공지능 모델인 Artificial Neural Network(ANN)에서 일부 연산을 컨볼루션 연산으로 대체한 이미지에 특화된 신경망 구조이다. 일반적인 CNN은 마지막 단의 완전 연결 계층을 이용하여 데이터를 분류하는데, 완전 연결 계층의 노드는 독립적인 일차원 벡터의 형태이기 때문에 공간적 정보는 버려지게 된다. FCN은 이를 해결하기 위해 완전 연결 계층 대신 컨볼루션 연산을 사용하는 구조이다. FCN을 이용하면 한 이미지 내에 분류 대상 객체가 여러 개 존재할 경우, 각 객체가 차지하는 영역을 상세하게 검출할 수 있다. 또한 노드의 개수가 고정되는 완전 연결 계층 대신 이차원 필터를 학습하는 컨볼루션 레이어를 사용하므로 입력 이미지의 크기에 관계 없이 학습 및 검출을 할 수 있다[5].

### 2-2 소리 분류 연구

기존의 소리 분류 연구는 대부분 딥러닝으로 수행되었다. 야생조류의 울음소리를 통해 분류 시스템을 제작하거나, 돼지 울음소리를 이용하여 돼지의 호흡기 질병을 탐지한 연구에도 딥러닝 모델이 사용되었다[6][7]. UrbanSound8K는 도시 환경음이 최대 4초 동안 녹음된 오디오 데이터 셋이다. Urbansound8K는 에어컨, 자동차 경적, 드릴, 엔진, 망치, 총, 사이렌, 길거리 음악, 강아지 짖는 소리와 아이들이 노는

소리의 총 10종류로 전체 8732개의 음원으로 구성되어있다 [8]. 본 논문에서는 UbanSound8K를 학습 데이터로 설정하여 딥러닝 학습을 진행했다.

### III. 제안하는 시스템 구성

본 논문의 프로젝트 구성도는 그림 1과 같다. OpenSSL(Open Secure Socket Layer)의 인증서를 통해 높은 보안을 가진 플랫폼으로 구현하고자 하였다. 적용된 알고리즘에는 파이썬의 Numpy, Keras, Tensorflow와 JavaScript의 Mozilla 등을 사용하였다. 웹 어플리케이션은 파이썬의 웹 프레임워크인 Flask를 이용하여 구현하였다. Flask는 경량 WSGI(Web Server Gateway Interface) 웹 애플리케이션 프레임워크로, 복잡한 애플리케이션으로 확장할 수 있다. 플랫폼 서버 관리는 PyCharm과 깃허브(GitHub)를 통해 수행하였다.

그림 2는 제안하는 플랫폼의 메인 페이지에서 보이는 세 가지 버튼이다. 첫 번째 버튼은 사용자의 기기에서 녹음된 오디오 파일을 딥러닝 신경망에 입력하여 특정 소리를 감지하는 기능을 실행하는 페이지로 연결된다. 두 번째 버튼은 사용자 기기에서 수집된 음성을 글자로 바꿔주는 음성-텍스트 변환 서비스를 제공하는 페이지로 연결된다. 세 번째 버튼을 누르면 청각장애인과 관련된 최신 뉴스가 정리된 페이지로 연결된다.

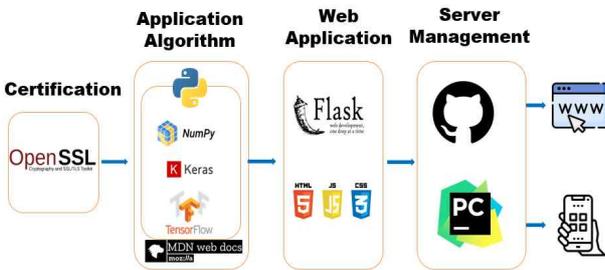


그림 1. 프로젝트 구성도  
Fig. 1. Project Structure

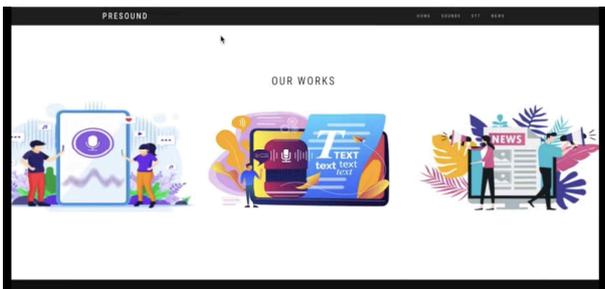


그림 2. 딥러닝을 이용한 청각장애인 위험방지 및 음성-텍스트 변환 서비스 메인 페이지  
Fig. 2. The main page of Risk Prevention and Speech-To-Text Service for the Deaf Using Deep Learning.

## IV. 소리 분류를 위한 딥러닝

### 4-1 딥러닝 구현에 사용된 프레임워크

딥러닝 알고리즘은 Tensorflow, Keras, Scikit-learn을 사용하여 구현하였다. Tensorflow는 구글이 2015년에 공개한 머신러닝을 위한 오픈 소스 라이브러리로, 머신러닝과 딥러닝을 손쉽게 구현할 수 있도록 설계되었다. Tensorflow는 유연하게 새로운 머신러닝 및 딥러닝 모델을 정의하고 데이터를 학습시키는 대중적인 라이브러리이다[9]. Tensorflow는 알고리즘 미분 혹은 자동 미분 기능을 내장하고 있기 때문에 사용자가 정의한 예측 모델에 대하여 경사 하강 기법을 수행할 수 있다[10].

Keras는 기계 학습 모델의 용이한 설계 및 구현을 위하여 고수준의 API(Application Programming Interface) 및 관련 클래스를 제공하고, 딥러닝 프레임워크인 Tensorflow를 백엔드로 사용한다. 따라서 Keras는 Tensorflow의 특수 기능을 모두 지원하며 동일한 코드로 CPU와 GPU에서 실행할 수 있어, 실험 속도를 개선할 수 있다[11]. Keras는 상위수준 래퍼로 실제적인 연산을 하지 않고, 백엔드에서 연산이 진행되어서, 편의성이 좋은 라이브러리이다 [12].

Scikit-learn은 파이썬의 머신러닝 라이브러리로, 다양한 분류, 회귀, 클러스터링 알고리즘을 제공한다. 더불어 데이터 전처리 및 조정, 모델 평가 등을 수행할 수 있다[13]. Scikit-learn 함수 train\_test\_split를 두 번 사용하여 불균형을 해결하고자 계층화하여 train set과 test set이 동일한 비율을 갖도록 하였고, 유효성 검사 및 테스트를 생성하였다.

데이터 전처리를 위해 Librosa로 데이터 세트의 불규칙성과 비정형 데이터를 정규화하였다. 오디오의 지속시간이 0초에서 4초이므로 생성된 Spectrogram을 패딩하며 모든 오디오를 동일한 길이로 만들었다. 모든 오디오 데이터를 Librosa와 기본 샘플링 속도 22050Hz를 사용하였다. 인간은 약 20000Hz까지 들을 수 있어 낮은 속도로 음악과 음성 데이터를 성공적으로 분석할 수 있다.

### 4-2 딥러닝 모델 제작

본 논문에서는 구글에서 제공하는 Colaboratory에서 딥러닝 구현 및 오디오 학습을 진행하였다. Colaboratory은 주피터 노트북(Jupyter Notebook) 개발환경으로 코랩(Colab)이라는 약어로도 불린다. 구글 드라이브에 접근하여 드라이브에 저장된 파일을 불러올 수 있고, 브라우저에서 파이썬 스크립트를 작성할 수 있다.

초기에는 CNN 모델을 사용하여 Conv2D 레이어 3개, Pooling 레이어와 Dense 레이어 각 2개 그리고 1개의 Flatten 레이어로 구성하였다.

Layer (type)	Output Shape	Param #
conv2d_20 (Conv2D)	(None, 38, 171, 16)	144
max_pooling2d_15 (MaxPoolin g2D)	(None, 19, 57, 16)	0
conv2d_21 (Conv2D)	(None, 18, 54, 32)	4128
max_pooling2d_16 (MaxPoolin g2D)	(None, 9, 27, 32)	0
conv2d_22 (Conv2D)	(None, 8, 24, 64)	16448
max_pooling2d_17 (MaxPoolin g2D)	(None, 4, 12, 64)	0
conv2d_23 (Conv2D)	(None, 3, 9, 128)	65664
global_average_pooling2d_5 (GlobalAveragePooling2D)	(None, 128)	0
dropout_3 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 10)	1290

=====  
 Total params: 87,674  
 Trainable params: 87,674  
 Non-trainable params: 0  
 =====

**그림 3. FCN 모델 요약**  
**Fig. 3. Summary of FCN Model**

하지만 해당 모델에 학습을 시킨 결과, 정확도는 약 65%로 수렴하였기 때문에 개선을 위하여 컨볼루션 레이어의 개수를 늘리고 각 레이어의 필터 크기를 모두 동일한 값이 아닌, 점차 높이는 방식으로 모델을 수정하였다. 또한, 데이터의 특징인 Mel Frequency Cepstral Coefficients(MFCC)를 추출할 때, 이전과 달리 구체적인 파라미터를 입력하여 더욱 분명한 특징 추출이 가능하도록 FCN 모델을 구성하였다. 여러 개의 레이어가 있는 Tensorflow에서 실행되는 Keras를 통해 FCN 모델을 구성하였다. 그림 3은 제작한 FCN 모델의 요약이다.

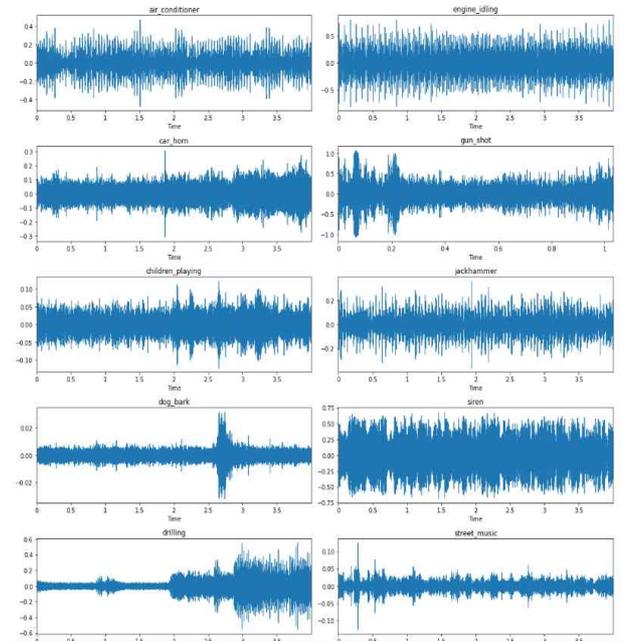
입력 이미지는 직사각형 모양(y축은 MFCC, x축은 시간)이므로 정사각형 필터를 사용하는 대신 직사각형으로 만들어 MFCC와 시간 차원의 상관관계를 더 잘 학습할 수 있도록 하였다. 소리 분류 신경망은 Conv2D 레이어 4개, Max Pooling 레이어 3개, Global Average Pooling 레이어 1개로 구성했다. 또한, 과적합을 방지하기 위하여 마지막 레이어 전에 Dropout을 추가 하였다. 컨볼루션 레이어는 입력 데이터에서 특징을 추출하여 특징맵을 얻는 역할을 하고, 풀링 레이어는 컨볼루션 레이어에서 증가된 차원을 감소시키는 레이어이다. Max pooling은 특징맵에서 2x3 사이즈에 해당되는 영역안의 최댓값만 뽑아서 특징맵의 크기를 줄여주고 Global Average Pooling은 Max pooling보다 더 급격하게 특징맵에서 특징(특징맵을 이루는 요소)의 수를 줄여서 특징맵을 1차원으로 만든다.

**4-3 학습 데이터 추출**

소리 분류를 위한 데이터 셋은 Urbansound8K를 구성하는 10개의 클래스에 대한 오디오 데이터 8732개로 설정하였다. 파이썬의 Librosa 라이브러리를 활용하여 오디오 데이터의 MFCC를 추출하여 학습 데이터를 준비하였다. 샘플링 속도는 8kHz에서 192kHz(대부분 44kHz, 48kHz)이고, 오디오의 길이는 0.0008초에서 4초(대부분 4초)까지이다. 그림 4는 학습한 데이터의 요약이다. salience는 소리의 감도 등급을, classID는 사운드 클래스를 숫자 식별자로 일상생활 속 발생하는 위험요인 10가지 소리를 나타낸다. 자동차 경적 소리와 사이렌 소리는 다른 클래스와 다르게 분류되지 않아 손실 함수 또는 오버 샘플링에 가중치 클래스를 포함하는 방식을 통해 탐색하였다. 데이터는 오디오 파일로 구성되므로 Librosa 라이브러리를 사용하여 오디오를 로드, 플롯한다.

	salience	classID	length	bitrate	channels	sample_rate
count	8732.000000	8732.000000	8732.000000	8.732000e+03	8732.000000	8732.000000
mean	1.347000	4.592877	3.603644	4.495311e+05	1.915369	48456.979272
std	0.476043	2.894544	0.980913	5.480813e+05	0.278348	15300.080707
min	1.000000	0.000000	0.000816	1.102500e+04	1.000000	8000.000000
25%	1.000000	2.000000	4.000000	3.528000e+05	2.000000	44100.000000
50%	1.000000	4.000000	4.000000	3.528000e+05	2.000000	44100.000000
75%	2.000000	7.000000	4.000000	5.292000e+05	2.000000	48000.000000
max	2.000000	9.000000	4.000000	4.515840e+07	2.000000	192000.000000

**그림 4. 학습 데이터 요약**  
**Fig. 4. Describe Training Data**



**그림 5. 학습 데이터 시각화**  
**Fig. 5. Data Visualization of Training Data**

그림 5는 각 클래스의 서로 다른 랜덤 샘플에 대한 과정을 시각화하였다. Short-Term FourierTransform(STFT), Mel-Spectrogram, MFCC는 오디오 신호를 처리하고, CNN과 같은 기계 학습 알고리즘을 위한 입력 데이터를 생성한다.

그림 6은 STFT를 시각화한 사진이다. STFT는 시간 정보 유실을 방지하기 위해 일정한 단위로 나누어 푸리에 변환을 적용한다. 일반적으로  $n\_fft$ 는 1/4 정도가 곱치도록 설정하므로 본 논문에서는 2048로 적용하였다.

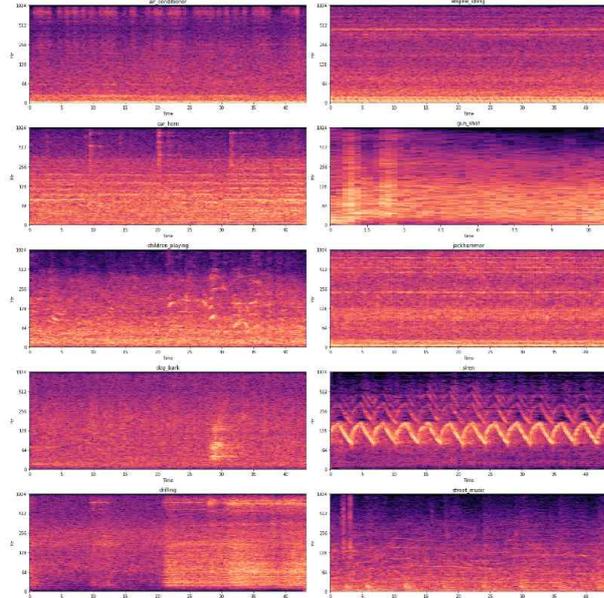


그림 6. STFT(로직 Y축) 시각화  
Fig. 6. Visualize STFT (Log Y Axis)

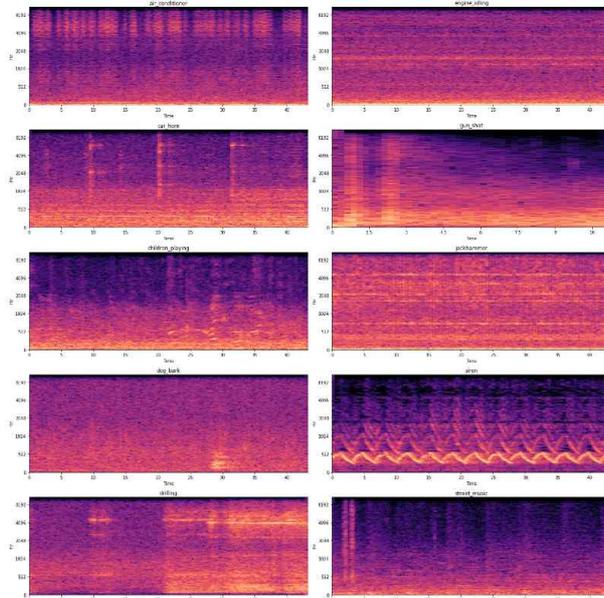


그림 7. Mel-Spectrogram 시각화  
Fig. 7. Visualize the Mel-Spectograms

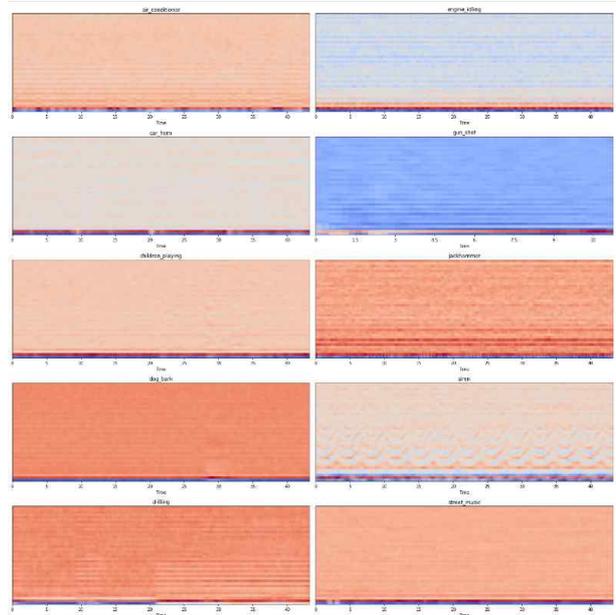


그림 8. MFCC 시각화  
Fig. 8. Visualize 40 MFCCs

그림 7은 다양한 주파수 크기를 다른 시간 단계에서 나타내는 Mel-Spectrogram을 시각화하였다. 주파수 크기는 인간이 오디오 신호를 인지하고 처리하는 방법에 기반하는 Mel-Spectrogram로 변환된다. 일부 다른 순열에 대한 그래프를 살펴본 후, 데이터 세트 클래스에 대한 신호가 올바르게 분류될 수 있을 정도로 다른 것을 관찰할 수 있다. MFCC는 음성데이터를 특징벡터화 해주는 알고리즘으로, 파이썬의 Librosa 라이브러리를 이용해서 간단히 추출할 수 있다[14]. Mel-Spectrogram의 특징에 대해 행렬을 압축해서 표현해주는 이산 코사인 변환 연산을 수행하면, MFCC를 출력할 수 있다. Mel-Spectrogram은 주파수끼리 상관관계가 형성되어 있는데, 이러한 상관관계를 없애는 역할 또한 수행한다. Mel-Spectrogram과 MFCC의 중요한 차이점은 Correlate와 De-Correlate이다. Mel-Spectrogram의 경우 주파수끼리 상관관계를 형성하고 있기 때문에 도메인이 한정적인 문제에서 더 좋은 성능을 보이고, MFCC의 경우는 상관관계가 없기 때문에 일반적인 상황에서 더 좋은 성능을 보인다. 그림 8은 MFCC를 시각화한 사진이다.

#### 4-4 오디오 학습

본 논문에서는 UrbanSound8K로부터 추출한 MFCC를 학습 대상으로 설정하였다. 그림 3과 같이 구성한 FCN 모델에 epochs는 200, batch size는 256으로 설정하여 학습을 진행하였다. 그 결과, 정확도는 약 91.07%로 측정되었다. 그림 9는 200번의 epochs에 따른 모델의 분류 정확도 그래프이다. 가로축은 epochs, 세로축은 분류 정확도이다.

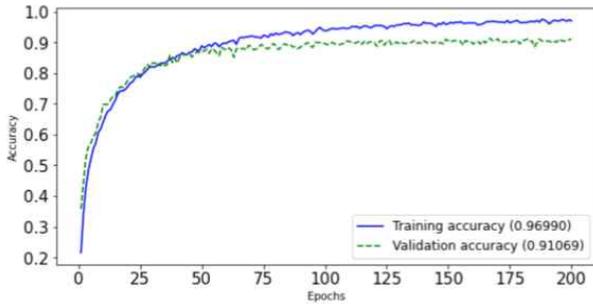


그림 9. 정확도 그래프  
Fig. 9. Visualize Accuracy Graph

### V. 음성-텍스트 변환 서비스

청각장애인은 대화할 때 상대방의 입 모양을 통해 말의 의미를 파악하는데, 마스크 등의 요인으로 입 모양을 볼 수 없는 상황에서는 곤란을 겪는다. 상대방이 수화를 알지 못한다면, 의사소통에 큰 어려움이 있을 것이다. 이와 같은 상황을 극복하고자, 플랫폼에 음성-텍스트 변환기능을 제공할 수 있도록 하였다. 음성-텍스트 변환은 음성을 글자, 숫자, 기호 등으로 변환하는 것으로 JavaScript를 사용하여 구현하였다. 사용자의 입력을 기다리다가, 사용자의 음성이 입력될 때 이를 인식하고 음성-텍스트 변환을 통해 입력된 음성을 텍스트로 변환한다[15].

기존에는 음성-텍스트 변환과 CNN을 위한 마이크를 아두이노 음성인식 키트를 이용해서 데이터를 입력받을 예정이었지만, Javascript의 Web Kit를 이용해서 목소리를 텍스트로 변환하여 사용자가 하드웨어를 따로 구비하지 않아도 사용자가 더욱 편리한 서비스를 이용할 수 있는 환경을 마련하였다.

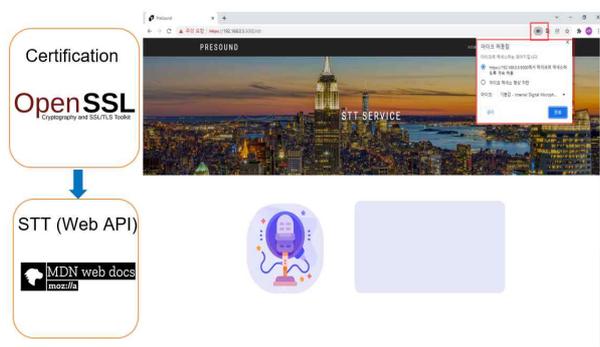


그림 10. 음성-텍스트 변환 서비스 마이크 인식  
Fig. 10. Speech-To-Text Service microphone recognition

음성-텍스트 변환을 간단하게 구현하는 방법에는 파이썬의 클래스인 Speech Recognition을 이용하는 방법과 JavaScript의 Mozilla를 사용하는 방법이 있다. 하지만 플랫폼에서 보여주고자 하는 음성-텍스트 변환 서비스는 그림 10처럼 같은 페이지에 결과값을 출력하는 것이 목적이지만 파

이쁜은 새 페이지에서 결과값이 출력되는 문제점이 있다. 이는 청각장애인과 소통을 원활히 한다는 목적에 맞지 않기 때문에 한 페이지 안에서 음성-텍스트 변환 서비스를 제공하기 위해 JavaScript의 Mozilla모듈을 사용하였다.

기존에 Mozilla를 이용하면 서비스를 배포할 때 인증되지 않은 페이지에서는 개인의 정보가 유출될 수 있기 때문에 홈페이지에서 자체적으로 마이크 사용을 차단한다. 이에 따라 OpenSSL의 인증서를 통해 홈페이지의 보안을 높임으로써 홈페이지 내에서도 마이크 사용이 가능하도록 하였다. 그림 10은 제안하는 서비스의 마이크 인식화면으로, 우측 상단에 마이크 사용이 허용되었음을 확인할 수 있다. OpenSSL은 오픈 소스 기반의 라이브러리로, 보안 통신 프로토콜인 SSL(Secure Sockets Layer)과 TLS(Transport Layer Security)을 구현하였다[16]. SSL은 암호 프로토콜로, 인터넷에서 일어나는 트랜잭션의 기밀을 보장하기 위해 기밀성, 무결성 및 인증 기능을 제공한다. TLS는 전송 계층 보안으로, 컴퓨터 네트워크에 통신 보안을 제공하기 위해 설계된 암호 규약이다. 이 규약은 TCP/IP 네트워크를 사용하는 통신에 적용되며, 통신 과정에서 전송계층 중단 간 보안과 데이터 무결성을 확보한다. 그림 11은 플랫폼에서 실제로 음성-텍스트 변환 기능이 실행되어 사용자의 음성에 대한 결과값이 출력된 화면이다.



그림 11. 음성-텍스트 변환 실행 장면  
Fig. 11. Execution scene of Speech-To-Text

### VI. Web Crawling

주제가 무분별한 뉴스에서 사용자가 청각 장애인에게 필요한 정보를 편리하게 정보를 얻을 수 있도록 하였다. 웹 크롤링을 통해 청각장애인을 위한 최신 이슈와 구직 정보 등 뉴스를 추출하여 제공함으로써 청각장애인을 위한 통합 플랫폼을 구현하고자 하였다. 빅데이터는 비정형 데이터로 수집에 제약이 존재하여, 이를 해결하기 위해 정보를 추출하고 변형하기 위해 웹 크롤링을 사용하였다. 빅데이터 분석을 목적으로 웹 크롤러는 수집 채널의 웹 페이지 구조에 특화된 수집 로직을 구현해야 한다. 웹 크롤링은 빅데이터 시대에 있어서 필수 불가결한 데이터 수집 방법론으로, 다양한 유용성을 갖는 기술의 일종이다[17].

```

@app.route('/news', methods=['GET', 'POST'])
def news():
    headers = {
        'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.86 Safari/537.36'
    }
    data = requests.get('https://www.fnnews.com/search?search_txt=청각장애인&page=0', headers=headers)
    soup = BeautifulSoup(data.text, 'html.parser')
    news = soup.select('#root > div.contents > div > div.wrap_left > div.wrap_artist.mt20 > div.inner_artist > ul > li')
    news_dict=[]
    for i in range(9):
        for new in news:
            a_tag = new.select_one('a')
            if a_tag is not None:
                url='https://www.fnnews.com'
                url=new.select_one('a:nth-child(1)')['href']
                image = new.select_one('a:nth-child(1) > span > img')['src']
                title = a_tag.select_one('strong').text.strip()
                desc = new.select_one('a:nth-child(2) > p').text.strip()
                date = new.select_one('em').text
                news_dict[str(url)]=
                {
                    'img':image,
                    'title':title,
                    'desc':desc,
                    'date':date
                }
    print("크롤링 완료")

    return render_template('news.html',news_dict=news_dict)

```

그림 12. 뉴스 크롤링 함수 코드  
Fig. 12. Code of new crawling function

웹 크롤링을 수행하는 프로그램인 웹 크롤러는 사이트에 저장되어 있는 정보들을 자동으로 읽고, 이를 저장하여 데이터베이스를 구축한다. 이러한 웹 크롤러는 BeautifulSoup, Selenium, Jsoup과 같은 웹 스크래핑 라이브러리를 이용하여 특정 웹 사이트에서 본문을 수집하였지만, 이때의 수집 로직은 웹 페이지 구조가 상이한 모든 수집 채널 각각에 맞게 개별적으로 구현되어야 한다[18].

파이썬의 BeautifulSoup을 이용하여 뉴스를 수집하고, 네이퍼에서 제공하는 API를 이용하여 기사를 수집하였다. BeautifulSoup은 HTML 코드를 파이썬이 이해하는 객체 구조로 변환하도록 Parsing을 진행하고, 의미 있는 정보만을 추출하도록 하는 라이브러리이다[19]. 요청과 응답 패킷은 header와 body로 구분되어 있고, header는 key와 value 형식으로 세팅되어 있다. User-Agent에는 브라우저 종류를, Referer에는 이전 페이지의 URL을, Accept-Language에는 응답받을 언어 및 Authorication의 인증정보를 입력해야 한다. request 라이브러리는 html에 찾고자 하는 부분이 있는 경우에 사용한다. Selenium 라이브러리는 원하는 데이터가 응답되는 HTML에 있지 않고, 이후 javascript를 통해 처리되는 경우 사용한다.

그림 12는 뉴스 크롤링 함수 코드이다. 한 페이지에 지정된 개수의 뉴스가 보이도록 for문을 원하는 개수만큼 반복 실행하며 원하는 뉴스를 추출한다. 관련된 뉴스의 상세 정보가 담긴 url, image, title, description, date를 추출하여 제공한다. 또한, 데이터를 실시간으로 크롤링하여 뷰로 가져오고, 원하는 뉴스를 클릭하면 ‘파이낸셜 뉴스’의 상세 페이지로 이동하도록 설계하여, 데이터베이스에 저장하지 않고도 원하는 서비스를 구현하였다.



그림 13. 웹 크롤링으로 청각장애인 관련 뉴스 제공  
Fig. 13. Provide new about the deaf using Web Crawling

그림 13은 뉴스 크롤링을 구현한 화면으로, 데스크탑, 태블릿, 모바일에서 사용 가능한 반응형으로 구현하였다. 크롤링한 데이터를 모두 데이터베이스에 저장한다면 서버에 부담을 줄 수 있고, 비용이 발생하지만, 제안하는 플랫폼은 사용자가 회원가입 하지 않고도 사용 가능하므로 데이터베이스의 부담을 줄이고, 개인정보 노출의 위험을 줄인 안전한 서비스를 제공한다.

이와 같은 웹 크롤링을 통해 청각장애인들을 위한 최신 소식을 알리는 서비스를 제공하고자 하였다. ‘파이낸셜 뉴스’에서 청각장애인에 관한 뉴스만을 크롤링하여 제공하므로 플랫폼의 사용자인 청각장애인이 원하는 정보를 편리하게 정보를 얻을 수 있다.

## VII. 결론

청각장애인은 사이렌 소리가 발생하거나 강아지가 짖어도 해당 소리의 발생을 감지하기 어렵다. 본 논문의 실험에서는 10개의 환경음에 대한 딥러닝 학습을 수행하였고, 약 91.07%의 분류 정확도를 가지는 모델을 도출하였다. 해당 분류 모델을 적용한 플랫폼을 통해 사용자의 주변에서 특정 소리가 발생했음을 파악한다면, 이전보다 사고에 신속하게 대처할 수 있다. 예를 들어 실내에서는 아이의 울음소리를 감지하여 아이의 상태를 빠르게 확인할 수 있고, 실외에서는 경적 소리를 감지하여 사용자 주변으로 접근하는 자동차를 피할 수 있다. 플랫폼에서 제공되는 상대방의 말을 문자로 출력하고 사용자가 입력한 문자를 소리로 출력하는 기능을 사용하면, 수화를 알지 못하는 상대방과 청각장애인 간의 원활한 의사소통이 가능하다. 사용자의 의견이 소리로 출력될 수 있기 때문에, 시각 장애인과의 의사소통 또한 할 수 있다. 위의 소리 분류 및 의사소통을 도와주는 기능과 더불어, 웹에서 제공하는 뉴스 및 구직 정보 등을 통해 최신 소식을 접할 수 있다. 이와 같이 본 논문에서 구현한 청각장애인을 위한 통합 플랫폼은 널리 활용될 수 있을 것으로 기대된다.

## 참고문헌

- [1] S. E. Mun, S. B. Jang, J. H. Lee, and J. S. Lee, "Trends of Machine Learning and Deep Learning Technology," *Journal of The Korean Institute of Communication Sciences*, Vol. 33, No. 10, pp. 49-56, October 2016.
- [2] T. Lin, T. Guo, K. Aberer, "Hybrid Neural Networks for Learning the Trend in Time Series," *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, pp. 2273-2279, August 2017.  
<https://doi.org/10.24963/ijcai.2017/316>
- [3] J. W. Kim, H. A. Pyo, J. W. Ha, C. K. Lee, J. H. Kim, "Deep learning algorithms and applications," *Communications of the Korean Institute of Information Scientists and Engineers*, Vol. 33, No. 8, pp. 25-31, August 2015.
- [4] S. W. Park, H. W. Kim, S. J. Lee, Y. S. Yun, E. S. Kim, J. H. Lim, Y. W. Lee, "Detection of Forest Disaster Areas Using High-resolution Satellite Images with Fully Convolutional Network," *Journal of the Association of Korean Photo-Geographers*, Vol. 28, No. 4, pp. 87-101, 2018.  
<https://doi.org/10.35149/jakpg.2018.28.4.006>
- [5] K. S. Lee, J. W. Na, J. D. Sohn, S. M. Sohn, and S. C. Lee, "Image Recognition Algorithm for Maintenance Data Digitization : CNN and FCN," *Transactions of the Korean Society for Noise and Vibration Engineering*, Vol. 30, No. 2, pp. 136-142, April 2020.  
<https://doi.org/10.5050/KSNVE.2020.30.2.136>
- [6] M. J. Kang, Y. S. Kim, H. Y. Shin and J. W. Park, "Development of the Deep Learning System for Bird Classification Using Birdsong," *Journal of Knowledge Information Technology and Systems*, Vol. 15, No. 2, pp. 195-203, April 2020.  
<https://doi.org/10.34163/jkits.2020.15.2.005>
- [7] M. K. Hong, H. S. Ahn, J. U. Lee, D. H. Park and Y. H. Chung, "Porcine Wasting Diseases Detection using Light Weight Deep Learning," *Proceedings of the Korea Information Processing Society Conference*, pp. 964-966, November 2020.  
<https://doi.org/10.3745/PKIPS.y2020m11a.964>
- [8] J. Salamon and J. P. Bello "Unsupervised feature learning for urban sound classification," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 171-175, April  
<https://doi.org/10.1109/ICASSP.2015.7177954>
- [9] B. S. Hwang, E. N. Huh, S. M. Kang, "Real-time log analysis system using Tensorflow," *2017 Korea Computer Science Congress Paper Collection of the Korean Information Science Society Conference*, pp. 672-674, June 2017.
- [10] J. Y. Kim, "Introducing Google TensorFlow," *Journal of The Korea Society of Computer and Information*, Vol. 23, No. 2, pp. 9-15, December 2015.
- [11] C. M. Song, K. H. Lee, "Applicability Evaluation for Discharge Model Using Curve Number and Convolution Neural Network," *Ecology and Resilient Infrastructure*, Vol. 7, No. 2, pp. 114-125, June 2020.  
<https://doi.org/10.17820/eri.2020.7.2.114>
- [12] S. Y. Kang, E. J. Im, "Ktune : autotuner for parallelizing backpropagation," *Proceedings of the Korean Information Science Society Conference 2017 Korean Software General Conference Papers*, pp. 1699-1701, December 2017.
- [13] J. Y. Kim, H. Sim, "Development of a Sign Language Learning Assistance System using Mediapipe for Sign Language Education of Deaf-Mutuality," *The Journal of the Korea institute of electronic communication sciences*, Vol. 16, No. 6, pp. 1355-1362, December 2021.  
<https://doi.org/10.13067/JKIECS.2021.16.6.1355>
- [14] S. M. Son, S. H. Lee, S. H. Kim, and S. W. Lee, "Android launcher using TTS & STT for blind," *Proceedings of Winter Conference on Korean Institute of Information Scientists and Engineers*, pp. 1713-1715, December 2015.
- [15] Z. Shiler and S. Dubowski, "Time optimal paths and acceleration lines of robotic manipulators," *26th IEEE Conference on Decision and Control*, pp.199-204, December 1987.  
<https://doi.org/10.1109/CDC.1987.272741>
- [16] W. H. Ahn and H. S. Kim, "Attacking OpenSSL Shared Library Using Code Injection," *Journal of KIISE : Computer Systems and Theory*, Vol. 37, No. 6, pp. 226-238, December 2010.
- [17] S. J. Kwon, J. H. Lee, and C. M. Lee, "A Study on the Legal Perception of Web Crawling in the Data Economy Era," *Korean Journal of Industry Security*, Vol. 11, No. 3, pp. 73-100, December 2021.
- [18] J. H. Kim and E. G. Kim, "WCTT: Web Crawling System based on HTML Document Formalization," *Journal of the Korea Institute of Information and Communication Engineering*, Vol. 26, No 4, pp. 495-502, April 2022.  
<https://doi.org/10.6109/jkiice.2022.26.4.495>
- [19] Ryan Mitchell, *Web Scraping with Python 2nd Edition*, O'Reilly Media Inc, pp. 6-32, 2018.

**정예진(Yae-jin Jeong)**



2017년~현재 재 : 강원대학교 IT대학 전기전자공학과 재학

※관심분야 : 웹(Web), UX/UI, 사물인터넷(IoT), 인공지능(AI) 등

**정윤아(Yun-A Jung)**



2019년~현재 재 : 강원대학교 IT대학 전기전자공학과 재학

※관심분야 : 인공지능(AI), 딥러닝, 기계학습, 심층 신경망, CNN 등

**김세하(Se-Ha Kim)**



2016년~현재 재 : 강원대학교 IT대학 전기전자공학과 재학

※관심분야 : 인공지능(AI), 딥러닝, 음성 향상, 파이썬, 음성인식 등

**김동회(Dong-Hoi Kim)**



2005년 : 고려대학교 전파공학과 (공학박사)

1989년 1월~1997년 1월: 삼성전자 전임연구원  
2000년 8월~2005년 8월: 한국전자통신연구원 전임연구원  
2006년 3월~현재 재: 강원대학교 IT대학 전기전자공학과  
2020년 6월~현재 재: 강원대학교 정보화본부장 등  
※관심분야 : 무선 네트워크 및 사물인터넷(IoT) 등