일반논문 (Regular Paper)
방송공학회논문지 제27권 제6호, 2022년 11월 (JBE Vol.27, No.6, November 2022)
https://doi.org/10.5909/JBE.2022.27.6.885
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

소리 데이터를 이용한 불량 모터 분류에 관한 연구

장 일 식[a)], 박 구 만[a)‡]

A Study on the Classification of Fault Motors using Sound Data

Il-Sik Chang[a)] and Gooman Park[a)‡]

요 약

제조에서의 모터 불량은 향후 A/S 및 신뢰성에 중요한 역활을 한다. 모터의 불량 구분은 소리, 전류, 진동등의 측정을 통해 검출한다. 본 논문에서 사용한 데이터는 자동차 사이드미러 모터 기어박스의 소리를 사용하였다. 모터 소리는 3가지의 클래스로 구성되어 있다. 소리 데이터는 멜스펙트로그램을 통한 변환 과정을 거쳐 네트워크 모델에 입력된다. 본 논문에서는 불량 모터 구분 성능을 올리기 위한 데이터 증강, 클래스 불균형에 따는 다양한 데이터 재샘플링, 재가중치 조절, 손실함수의 변경, 표현 학습과 클래스 구분의 두 단계 분리 방법 등 다양한 방법을 적용하였으며, 추가적으로 커리큘럼 러닝 방법, 자기 스페이스 학습 방법 등을 Bidirectional LSTM Attention, Convolutional Recurrent Neural Network, Multi-Head Attention, Bidirectional Temporal Convolution Network, Convolution Neural Network 등 총 5가지 네트워크 모델을 통하여 비교하고, 모터 소리 구분에 최적의 구성을 찾을 수 있었다.

Abstract

Motor failure in manufacturing plays an important role in future A/S and reliability. Motor failure is detected by measuring sound, current, and vibration. For the data used in this paper, the sound of the car's side mirror motor gear box was used. Motor sound consists of three classes. Sound data is input to the network model through a conversion process through MelSpectrogram. In this paper, various methods were applied, such as data augmentation to improve the performance of classifying fault motors and various methods according to class imbalance were applied resampling, reweighting adjustment, change of loss function and representation learning and classification into two stages. In addition, the curriculum learning method and self-space learning method were compared through a total of five network models such as Bidirectional LSTM Attention, Convolutional Recurrent Neural Network, Multi-Head Attention, Bidirectional Temporal Convolution Network, and Convolution Neural Network, and the optimal configuration was found for motor sound classification.

Keyword : MelSpectrogram, Data Augmentation, Class Imbalance, Sound Classification

a) 서울과학기술대학교 나노IT디자인융합대학원(Graduate School of Nano IT Design Fusion, Seoul National University of Science and Technology)
‡ Corresponding Author : 박구만(Gooman Park)
E-mail: gmpark@seoultech.ac.kr
Tel: +82-2-970-6430
ORCID: https://orcid.org/0000-0002-7055-5568
· Manuscript June 17, 2022; Revised August 19, 2022; Accepted September 29, 2022.

Ⅰ. 서 론

데이터셋의 종류에 따라 특정 클래스의 발생 확률이 현저히 낮은 경우 제조 공정의 경우 불량 데이터를 구하는 것은 굉장히 어려운 일이다. 따라서 제조 분야에서는 이상치 탐지(Anomaly detection)라는 기법을 이용하여 정상과 불량을 판정하기도 한다. 비정상 데이터의 수가 현저히 작을 경우가 아니면 지도 학습을 통한 모델을 설계한다. 또한 정상 데이터만 가지고 있거나 정상데이터만 많이 보유한 경우에는 비지도 학습 혹은 반지도 학습을 통하여 모델을 설계한다. 본 논문에서는 정상 및 비정상 데이터가 모두 부족하고, 클래스 불균형도 존재하지만 불량 데이터의 클래스의 불균형이 현저히 작지 않기 때문에 지도학습으로 모델을 설계하였다. 소리를 이용한 분류는 단순 소리 분류뿐 아니라 소리를 이용한 감정 분류 문제 등 다양하게 연구되고 있다. 소리 데이터를 학습시키기 위해서 사용되었던 특징에는 MFCC(Mel-Frequency Cepstral Coefficient), 멜스펙트로그램(MelSpectrogram) 등을 사용하여 딥러닝 학습의 입력으로 사용된다. 소리 기반 분류는 CNN(Convolutional Neural Network)[1][2], LSTM, CNN + LSTM(Long Short-Term Memory)[3][4], Transformer[5]를 사용하여 분류하는 다양한 연구가 진행되었다. 본 논문에서 사용하는 소리 데이터는 자동차의 사이드미러 기어박스의 모터 소리를 사용한다. 좋은 학습을 위해선 다량의 학습 데이터 및 라벨링 작업이 필요하다. 이러한 라벨이 없이 학습하는 방법으로 자기 주도학습을 통한 전이 학습[6][7]의 연구도 활발하다. 하지만 본 논문에서 사용되는 데이터 셋은 데이터 셋 크기가 작고, 클래스별 불균형이 있으며, 정상과 비정상간의 데이터가 대부분 유사하고 일정 구간에서만 다른 특징을 나타내고 있어, 기존 연구에서 다루는 소리의 데이터와 다른 성향을 보인다. 본 논문에서는 Bidirectional LSTM Attention[8], CRNN(Convolutional Recurrent Neural Network)[9], Multi-Head Attention[10], BTCN(Bidirectional Temperal Convolution Network)[11], CNN(Convolution Neural Network)[12] 등 총 5가지 네트워크 모델을 사용하여 데이터 증강, 클래스 불균형에 따는 다양한 방법으로 데이터 재샘플링, 재가중치 조절, 손실함수의 변경, 표현 학습과 클래스 구분 두 단계로 분리하는 방법 등 다양한 방법을 적용하였으며, 추가적으로 커리큘럼 학습 방법[13]과 자기 페이스 학습 방법[14] 등의 방법을 적용하여 효과적인 데이터 불균형 상태에서의 불량 모터 구분 방법에 대해 비교 연구하였다. 본 논문의 구성은 본론에서 데이터 수집, 데이터 증강, 데이터 불균형을 개선하기 위한 방법, 본 논문에서 사용된 네트워크 모델, 추가적인 커리큘럼 학습 및 자기 페이스 학습 방법 적용 후 실험을 통해 분석 결과를 제시한 후 결론 및 향후 연구에 대하여 기술하였다.

Ⅱ. 본 론

1. 데이터 수집

사이드미러 기어박스는 총 232개로 모터 소리는 22,050Hz 샘플링 주파수로 2번 녹음하여 10초간의 소리를 측정하였다. 자동차 사이드미러 기어박스의 모터 소리 데이터의 특성상 소리는 모터의 접히는 소리와 펴는 소리 2가지의 소리가 측정된다. 본 논문에서는 모터당 접는 소리와 펴는 소리를 독립적으로 구분하고, 실제 학습에 사용된 소리의 시간

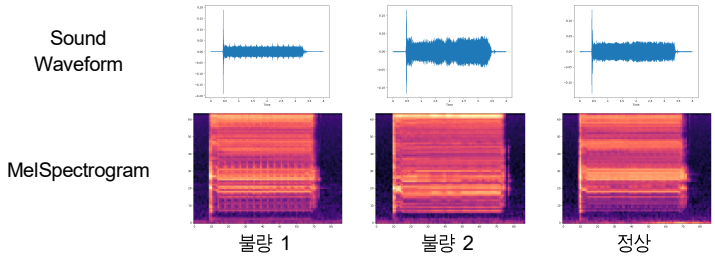

그림 1. 각 클래스 항목별 소리 파형과 멜스펙트로그램
Fig. 1. Sound waveform and MelSpectrogram for each class item

은 4초를 사용한다. 총 232개의 기어박스이기 때문에 모터의 소리 데이터 셋은 총 928개의 데이터 셋이 존재한다. 클래스는 총 3개로 구분하였다. 첫 번째 클래스는 모터의 웜 기어 샤프트가 기구적으로 잘못 정렬되었을 경우 발생하는 소리이다. 불량 1 클래스로 정의한다. 두 번째 클래스는 모터가 초기에 움직일 경우 샤프트가 벽에 부딪히면서 발생하는 소리이다. 불량 2 클래스로 정의한다. 세 번째 클래스는 정상 클래스를 나타낸다. 그림 1은 3가지 클래스에 대한 소리 파형 및 멜스펙트로그램으로 변환된 결과를 나타낸다.

각각의 클래스의 수는 정상 클래스 654개, 불량 1 클래스는 248, 불량 2 클래스는 26개로 총 데이터 셋의 크기가 부족하고, 클래스 불균형이 심하다. 또한 사람이 직접 청각에 의존하여 라벨링을 하였기 때문에 그라운드 트루스(Ground Truth)가 명확하다고 하기에 한계가 있다. 데이터 라벨링은 모터 기어박스 관련 전문가와 비전문가 5명으로 총 6명이 라벨링을 하였다. 전문가의 결과를 정답으로하고 비전문가와의 정답의 결과에 따라 총 3단계로 어려움의 단계로 구분하였다. 같은 모터의 데이터는 학습 데이터와 테스트 데이터가 서로 섞이지 않게 하고 총 10개의 폴드로 구성되어 있다.

2. 데이터 증강

부족한 데이터 셋을 보완하기 위하여 소리 데이터에 다양한 증강 기법이 존재한다. 데이터 증강 방법으로는 소리 데이터를 직접 변경하는 direct transformation, slicing window, window warping, flipping, warping+ensemble, noise injection, label expansion[15]등의 방법과 스펙트로그램(Spectrogram)의 특성값을 변경하는 SpecAugment[16], Mix SpecAugment[17], SpecSwap[18], SpecAugment++[19]등의 다양한 방법이 있다. 본 논문에서는 두 가지 방법을 모두 사용하여 데이터 증강을 하였다. 소리 데이터에 대한 증강 방법으로는 소리 데이터 일정 구간을 자르는 방법, 소리 데이터의 시작 구간 변경, 소리 데이터의 위치를 지정하여 해당 위치에서 랜덤하게 크기를 정하고 각 점에 대하여 스플라인 보간을 한 후 소리 데이터와 곱하는 크기에 대한 보간, 소리 정보에 평균이 0이고 표준편차가 1인 가우시안 정규분포의 값에 일정크기의 값을 더해서 입력데이터에 더하는 즉 노이즈를 추가하는 방법 등을 사용한다. 스펙트로그램으로 변경된 특징 벡터에 대한 데이터 증강 방식으로는 SpecAugment 방법에서 사용된 시간 마스킹, 주파수 마스킹 방법을 사용하였으며, 마스킹 영역은 평균값으로 적용하였다. 또한 SpecAugment++ 방법에서 사용된 혼합 마스킹, 커팅 마스킹 방법을 사용하였다. 실제 실험에서 소리 데이터에 대한 증강 및 멜스펙트로그램을 이용한 학습을 할 경우 매번 소리 데이터의 변환 과정이 필요하기 때문에 많은 학습시간을 필요로 하였다. 그리하여 미리 학습 데이터에 대하여 10번의 소리 데이터에 대한 증강 후 멜스펙트로그램으로 변환하여 파일로 저장하여 학습 데이터로 사용하였다.

3. 데이터 불균형

데이터 불균형을 해결하기 위한 방법은 크게 3가지로 분류할 수 있다. 첫 번째는 재샘플링으로 각 클래스 별로 데이터가 고르지 않는 문제를 샘플링 방법으로 해결하는 방법이다. 적은 클래스의 데이터를 중복하는 오버 샘플링 방법과 많은 클래스의 데이터를 제거하는 언더 샘플링 방법이 대표적이다. Square-root 샘플링, 점진적으로 균형 샘플링[20] 방법 등을 들 수가 있다. 두 번째는 재가중치를 적용하는 방법으로 클래스의 데이터의 크기에 반비례하는 가중치를 주는 방식이다. 또한 샘플의 개수를 동일하게 맞추는 방법이 아닌 Effective Number 라는 개념을 정의하고, Effective Number에 반비례하게 가중치를 주는 방법[21]등을 들 수가 있다. 세 번째는 클래스 균형 손실 함수를 이용하는 방법이다. 객체 검출에서 제안한 것으로서 어렵거나 오분류되는 경우에 대해 더 큰 가중치를 주는 Focal Loss[22], 적은 데이터를 가진 클래스에 더 큰 마진을 가지게 하는 LDAM(Label Distribution Aware Margin) Loss[23]등이 있다. 다른 방법으로 표현 학습과 클래스 구분을 두 단계로 분리하여 데이터 불균형의 문제를 해결하는 디커플링 방법[20]이 있다. 본 논문에서는 재샘플링 방법으로 모든 클래스에 대해 동일한 추출 확률을 주는 방법, 오버 샘플링으로 각 클래스 수에 반비례하도록 데이터를 샘플링하는 방법, Square-root 샘플링, 점진적으로 균형 샘플링등의 방법을 실험하였고, 재가중치 방법으로 Effective Number를 통

한 방법, DRW(Deferred Re-Weighting) 방법을 사용하였다. DRW는 첫번째 모든 샘플에 대하여 동일 가중치로 학습을 하고, 두 번째 스테이지에서 Effective Number를 통한 방법으로 학습을 하는 방식이다. 손실함수는 소프트맥스 손실함수, Focal Loss, LDAM Loss 방법을 사용하였다. 또한 표현학습과 클래스 구분을 두 단계로 분리하여 실험하고, 해당 방법에서 사용된 NCM(Nearest Class Mean classifier)의 성능의 가능성을 보고 메트릭 학습에서 Angular 마진이라는 개념을 사용한 Sphereface[24], Cosface[25], Arcface[26] 손실함수를 실험하였다.

4. 네트워크 모델

본 논문에서 사용한 데이터는 사이드미러 기어박스의 모터 소리 데이터로 멜스펙트로그램으로 변환 후 특징 벡터를 네트워크 모델의 입력으로 사용하였다. 소리 데이터는 시계열 데이터이므로 시계열 데이터에 사용되는 다양한 네트워크 모델을 적용하였다. 또한 멜스펙트로그램으로 변환된 특징 벡터가 주파수 측과 시간축으로 2차원 특성을 가지고 있기 때문에 이미지 처리에 많이 사용하는 CNN을 사용하였다. 네트워크 모델은 Bidirectional LSTM Attention, CRNN, Multi-Head Attention, BTCN, CNN 등 총 5가지 네트워크 모델을 사용하였다. 변환된 특징 벡터의 크기는 주파수 측은 64, 시간축은 87의 크기를 갖는다. 모든 네트워크 모델의 마지막은 Dense 레이어를 갖고 최종단에 3개의 클래스로 구분하는 모델을 갖는 구조이다. 그림 2는 Bidirectional LSTM Attention 구조를 나타낸다. LSTM의 잠재 벡터의 크기는 64를 갖는다.

그림 3은 CRNN 구조를 나타낸다. Conv Block은 2D Convolution, 배치 정규화, Relu, 커널 크기 2, 스트라이드 크기 2의 맥스풀링의 구조이다. LSTM의 잠재 벡터의 크기는 64를 갖는다.

그림 4는 Multi-Head Attention 구조를 나타낸다. Multi-Head Attention은 트랜스포머 구조의 인코더만 사용하는 구조로 특징 벡터 입력의 크기는 멜스펙트로그램의 주파수 크기인 64로 설정하고, 멀티 헤드의 수는 4개로 설정하였다. 피드포워드 네트워크의 차원은 512로 설정하였다. 활성화 함수는 Relu를 사용하였고, 드랍아웃은 0.4의 값을 가진다.

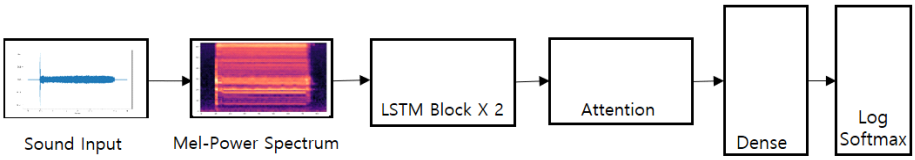

그림 2. Bidirectional LSTM Attention 구조
Fig. 2. Bidirectional LSTM Attention Structure

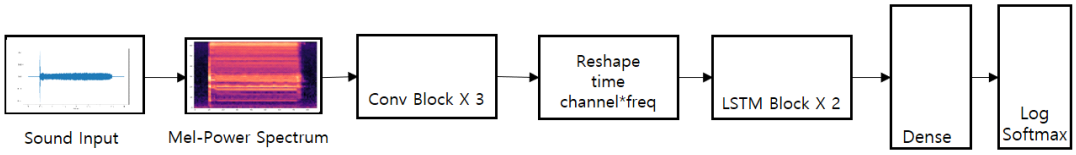

그림 3. CRNN 구조
Fig. 3. CRNN Structure

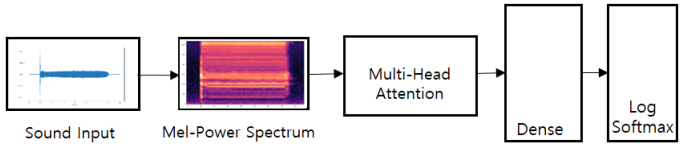

그림 4. Multi-Head Attention 구조
Fig. 4. Multi-Head Attention Structure

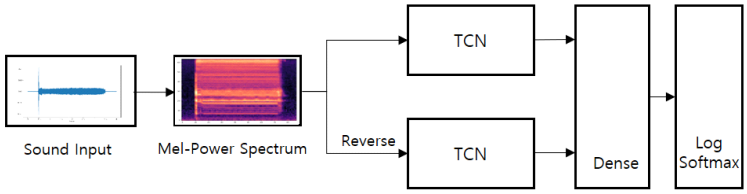

그림 5. BTCN 구조
Fig. 5. BTCN Structure

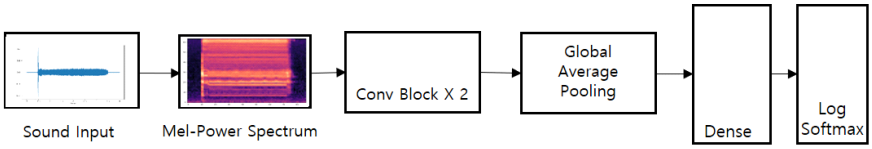

그림 6. CNN 구조
Fig. 6. CNN Structure

그림 5는 BTCN 구조를 나타낸다. TCN 구조의 커널 크기는 7이고, 잠재 벡터의 채널은 64, 팽창(Dilation) 레벨은 6으로 설정하였다.

그림 6은 CNN 구조를 나타낸다. 커널 크기가 3, 스트라이드 1인 컨볼루션, 배치 정규화, 활성화 함수로 Relu, 커널 크기 2, 스트라이드 2인 최대 풀링의 구조를 2개를 사용하였다. 첫 번째 출력 채널수는 32, 두 번째 출력 채널수는 64로 설정하였다.

5. 커리큘럼 학습 및 자기 페이스 학습

소리 데이터 라벨링은 모터 관련 전문가와 비전문가 5명으로 총 6명이 라벨링을 하였다. 하지만 라벨링된 결과가 전문가와 비 전문가간에 상이한 결과를 나타냈다. 전문가의 결과를 정답으로 하고 비전문가와 정답의 비율에 따라 총 3단계로 어려움의 단계로 구분하였다. 비전문가와의 결과와 4개 이상이 같으면 A, 비전문가와 결과가 2개 이상이면 B, 그 외는 C로 구분하였다. A는 766, B는 70, C는 92개로 전문가가 작성한 라벨링과 비전문가가 작성한 라벨의 차이가 큰 것을 볼 수 있다. 본 논문에서는 이러한 학습 데이터의 난이도를 구분함으로써 커리큘럼 학습에 적용하여 실험을 하였다. 커리큘럼 학습은 처음에는 쉬운 샘플을 모델에 적용하고, 점차 학습을 진행하면서 어려운 데이터를 네트워크 모델에 적용하는 방식이다. 하지만 커리큘럼 학습에서 어려운 데이터를 사용자가 직접 라벨링해야 한다는 문제점이 존재한다. 이러한 사전지식이 없는 데이터를 가지고 손실값이 크면 더 어려운 데이터로 가정하고 학습 단계가 진행할수록 학습에 포함하는 방식인 자기 페이스 학습 방법을 적용하였다. 본 논문에서 사용한 방법은 미니 배치단위로 손실값을 계산하고 초기 시작시점에서 80%의 데이터만 학습에 사용하고 점차 에폭이 증가하면서 학습 데이터를 100%까지 학습에 포함하는 방식을 사용하였다.

6. 실험

본 논문의 모든 실험은 학습 데이터와 테스트 데이터가 서로 섞이지 않도록하여 총 10개의 폴드로 구성되어 있고, 학습 데이터는 9개의 폴드 데이터를 사용하고, 테스트 데이터는 1개의 폴드 데이터를 사용하였다. 실험 결과는 총 5개의 서로 다른 폴드의 결과에 대한 평균이고, 학습은 총 50번의 에폭으로 하였다. 일반적으로 사용하는 소리 데이터셋으로 UrbanSound8K[27], RAVDESS[28], IRMAS[29]등의 3가지 데이터셋을 추가 실험하였다. 3가지 데이터셋은 소리 소리데이터와의 특성이 다름을 보이기 위해 데이터 불균형 해소를 위한 방법에 사용하여 모터의 소리 데이터와 비교하였다. UrbanSound8K는 도시 영상에서 8,732개의 사운드 데이터를 총 10개의 클래스를 가지고 너무 적은 데이터는 제외하고 3초간의 데이터를 사용하였다. IRMAS는 악기

표 1. 데이터 셋의 각 클래스 수
Table 1. Number of each class in the Dataset

Cls Num / Dataset	1	2	3	4	5	6	7	8	9	10	11
Our	248	26	654								
UrbanSound8K	997	227	987	721	841	982	61	861	901	1000	
IRMAS	388	505	451	637	760	682	721	626	577	580	778
RAVDESS	95	139	175	168	126	179	115	186			

소리 데이터 셋으로 총 6,705개의 3초간격 오디오 파일로 구성되어 있으며 총 11개의 클래스를 가지며, 마지막 RAVDESS은 감성 음성 오디오 데이터셋으로 1,440개의 파일로 구성되며, 8개의 클래스를 갖는다. 표 1은 데이터 셋의 각 클래스 수를 나타낸다.

6.1. 데이터 증강

데이터증강에 따른 실험은 세가지 방법을 사용하였다. 첫 번째는 소리 데이터에 대한 증강을 하기 위해선 소리 데이터를 먼저 증강한 후 멜스펙트로그램으로 변경 후 변경된 특징 벡터는 증강하지 않고 네트워크 모델의 입력으로 사용한다. 두 번째 방법은 소리 데이터 증강을 하지 않고 멜스펙트로그램으로 변환한 특징 벡터만 증강하는 방식을 사용한다. 세 번째 방법은 소리 데이터 증강 및 특징 벡터 증강 모두를 사용하는 것이다. 네크워크 모델의 입력은 초기 z-점수 정규화를 하고, 주파수의 크기는 64이고, 손실함수는 소프트맥스를 사용하였다. 본 논문에서 성능 지표로 사용하는 WA(Weight Accuracy)는 각 클래스별 정확도를 계산 후 클래스의 수로 나눈 평균값이다. 그림 7은 두 번째 방법으로 모터의 소리 데이터셋을 입력으로하여 특징 벡터만 증강한 결과이다. 특징 벡터 증강의 마스킹 방법의 설정은 마스킹 개수는 2개, 마스킹의 길이는 최대 전체 길이의 5%로 설정하였다. none은 불균형 데이터를 그대로 사용한 방법이고, over는 각 오버샘플링 방법을 의미한다. a1은 시간 마스킹, 주파수 마스킹 방법만 사용, a2는 혼합 마스킹, 커팅 마스킹 방법만 사용, a3은 두 가지 모두 사용하였을 때 방법이다. 네트워크 모델은 brnn_att(Bidirectional LSTM Attention), cnn(CNN), crnn(CRNN), mt_att(Multi-Head Attention), bi_tcn(BTCN)을 의미한다.

그림 7에서 특징 벡터 증강이 a3일 경우 안정적인 결과를 보인다. 그림 8은 모터의 소리 데이터셋을 입력으로하여 데

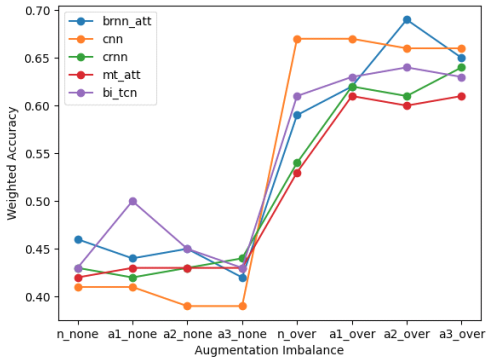

그림 7. 특징 벡터 증강
Fig. 7. Feature Vector Augmentation

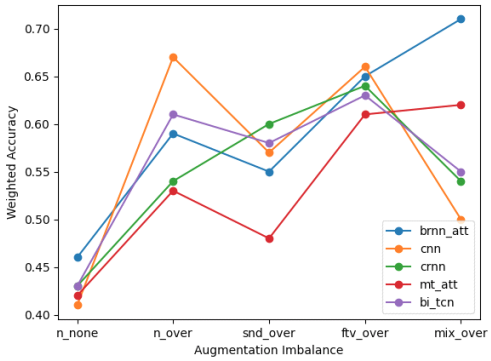

그림 8. 데이터 증강 결과
Fig. 8. Data Augmentation Result

이터 증강에 사용한 세 가지 방법을 적용한 결과이다. 세 번째 방법의 특징 벡터 증강방법은 특징 벡터 증강의 a3 방법을 사용하였다. snd는 소리 데이터 증강을 의미하고, ftv는 특징 벡터 증강, mix는 소리와 특징 벡터 증강 방법을 모두 적용했을 경우의 결과이다. 소리 데이터 증강은 소리 데이터 일정 구간을 자르는 방법의 확률은 0.1, 구간의 최대 길이는 10%, 소리 데이터 크기 증강 구간은 4개, 크기는 최대 0.2로 설정하였다. 노이즈 증강 확률은 0.2, 크기값은

0.05로 설정하였다.

그림 8에서 특징 벡터의 증강 방법이 가장 안정적인 결과를 나타내기 때문에 본 논문에서는 특징 벡터의 증강으로 이후 실험을 진행하였다. 소리 데이터에 대한 증강의 결과가 좋지 못한 것은 모터 소리 데이터가 대부분 유사하고, 일부 구간에서 서로 다른 특징을 가지기 때문에 소리 자체의 증강 방법은 좋지 못한 결과를 나타내는 것으로 보인다.

6.2 데이터 불균형

데이터 불균형 해소를 위한 다양한 방법으로 실험하였다. 그림 9는 네트워크 모델별 재샘플링 및 재가중치 방법에 대한 결과를 나타낸다.

그림 9에서 i1 : 모든 클래스에 대해 동일한 추출 확률을 주는 방법, i2 : 오버 샘플링, i3 : DRW, i4 : 점진적으로 균형 샘플링등의 방법, i5 : Effective Number, i6 : Square-root을 의미한다. 모터의 소리 데이터는 i2, i5 방법이 좋은 성능을 나타냄을 볼 수 있고, Bidirectional LSTM Attention의 네트워크 모델이 성능이 높게 나타나는 것을 볼 수 있다. 하지만 다른 데이터 셋의 경우 재샘플링 및 재가중치에 대한 결과가 크게 차이가 없고 네트워크 모델의 경우 CRNN의 경우가 성능이 높게 나타나는 것을 볼 수 있다. Urban-Sound8K의 경우 역서 데이터 불균형이 심하지만 재샘플링 및 재가중치에 효가가 크지 않았다. 본 논문의 모터 데이터의 크기가 작고, 데이터의 성향이 상이하기 때문에 이러한

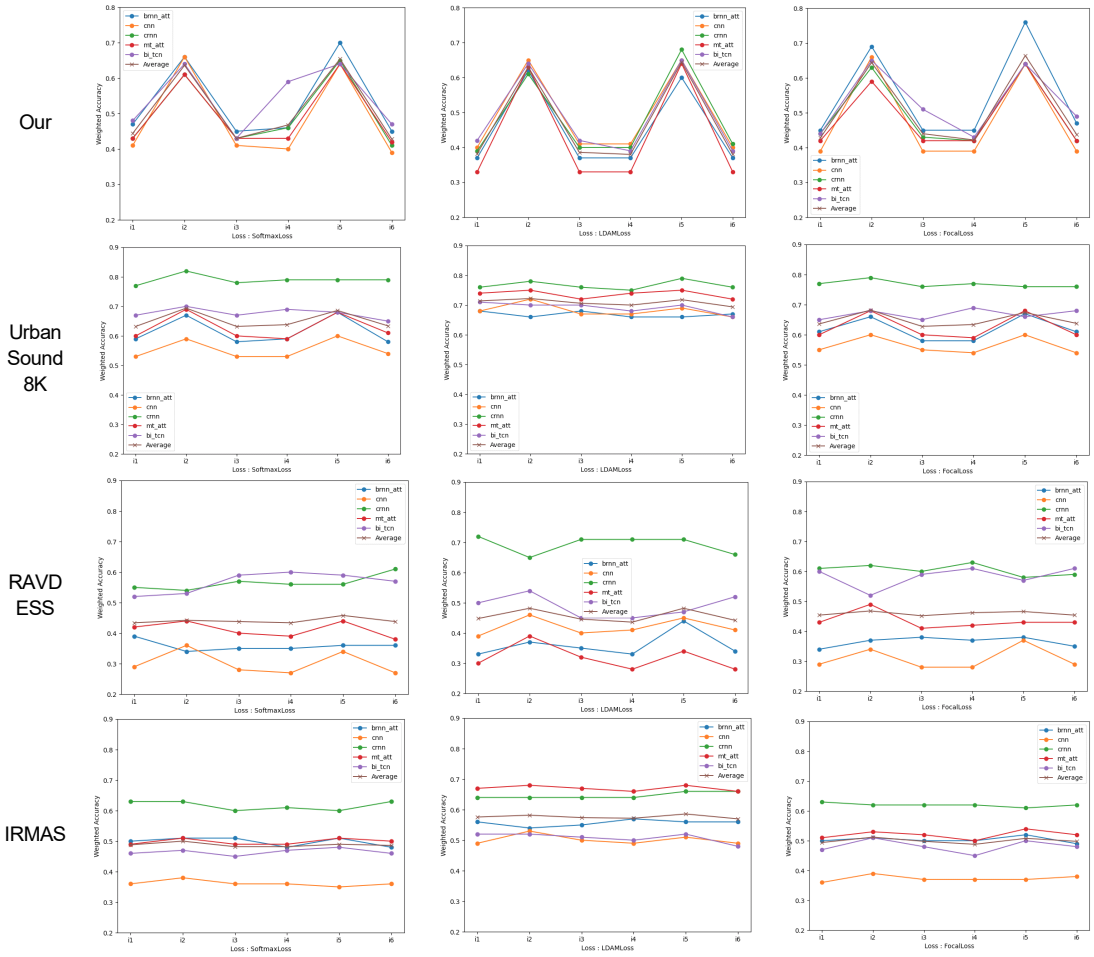

그림 9. 네트워크 모델 및 손실함수의 재샘플링 및 재가중치 방법에 대한 결과
Fig. 9. Results on resampling and reweighting methods of network models and loss functions

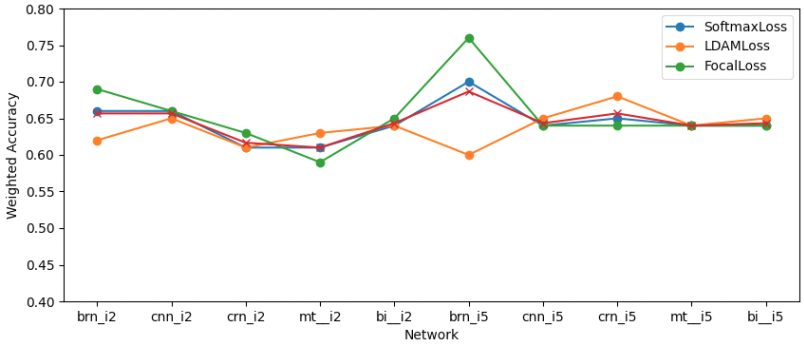

그림 10. 오버 샘플링 및 Effitive Number 방법에 대한 손실함수 결과
Fig. 10. Loss Function Results for Oversampling and Effitive Number Methods

결과를 보이는 것 같다. 그림 10은 모터의 소리 데이터셋을 입력으로하여 i2, i5 방법일 경우 손실함수에 따른 결과를 그림으로 나타낸 것이다.

Bidirectional LSTM Attention, Focal Loss 손실함수, Effective Number 방식일 경우 가장 좋은 성능을 보였다. 그림 11은 모터의 소리 데이터셋을 입력으로하여 메트릭 러닝에서 Angular 마진이라는 개념을 사용한 Sphereface, Cosface, Arcface 방법을 사용한 결과를 나타낸다. a는 Arcface, c는 Cosface, s는 Sphereface를 의미한다. 메트릭 학습의 결과는 네트워크 모델의 마지막 특징벡터의 거리로 클래스를 판별한다. 메트릭 학습을 사용한 방법에서는 좋은 성능이 나오지 못함을 볼 수 있다.

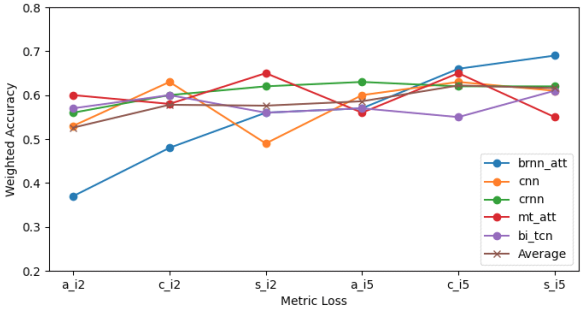

그림 11. Metric 학습 방법을 적용한 결과
Fig. 11. Results of applying the metric learning method

그림 12는 모터의 소리 데이터셋을 입력으로하여 표현 학습과 클래스 구분을 두 단계로 분리한 방법으로 디커플링한 방법을 적용한 결과를 나타낸다. d1 : End to End 방법, d2 : NCM, d3 : τ-normalized, d4 : LWS(Learnable weight scaling)[20]을 나타낸다. 실험은 모든 경우의 수를 하였으나 좋지 못한 결과는 그림에 표시하지 않았다.

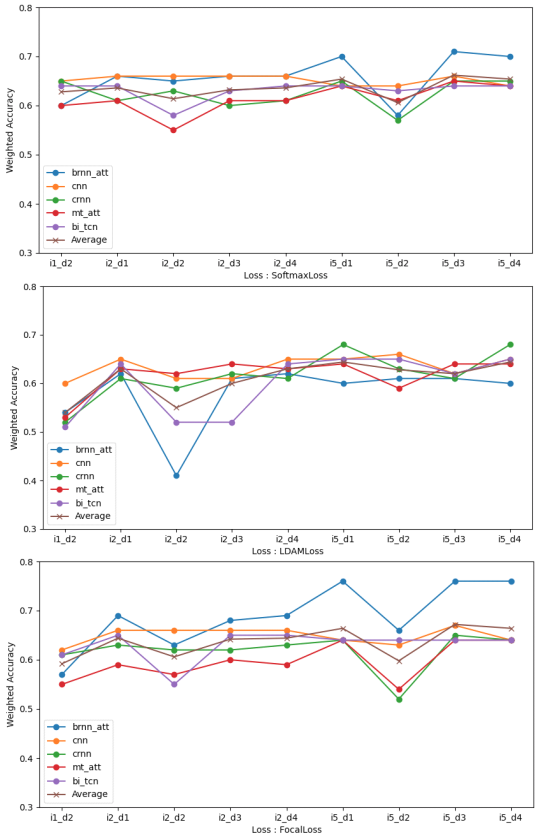

그림 12. 디커플링 방법의 결과
Fig. 12. Results of the decoupling method

그림 13은 성능이 가장 좋게 나온 Effective Number 방식

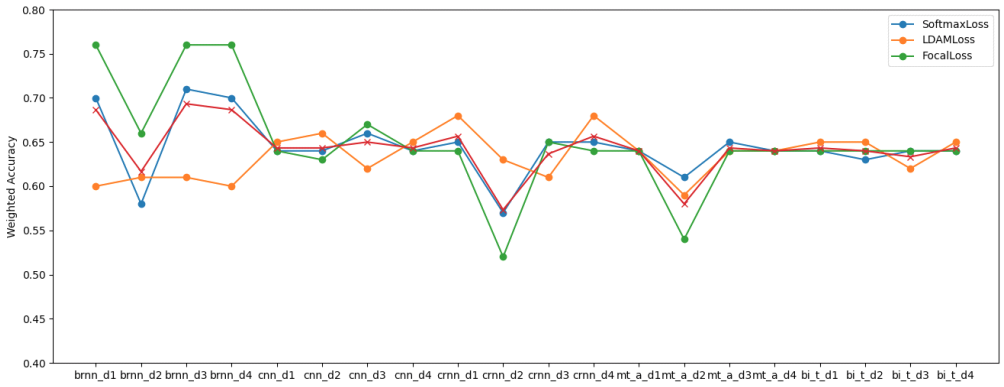

그림. 13. Effective Number 방식을 적용한 경우의 손실함수 결과
Fig. 13. Loss function result when effective number method is applied

에서의 손실함수에 따른 결과를 나타낸다. 하지만 End to End 방법과 큰 차이가 없기 때문에 디커플링 방식이 크게 효과적이진 않은 것을 볼 수 있다.

6.3. 커리큘럼 학습 및 자기 페이스 학습

본 논문에서는 학습 데이터의 난이도를 3가지로 나누었다. 3가지의 난이도에 대하여 10 에폭마다 어려운 학습 데이터를 추가로 학습하는 방법으로 실험을 하였다. 그림 14는 커리큘럼 학습에 대한 각각의 손실함수를 적용하였을 경우의 결과이다.

자기 페이스 학습은 미니 배치단위로 손실값을 계산하고 손실값이 크면 어려운 문제라 가정하고 초기 시작 시점에서 80%의 데이터만 학습에 사용하고 에폭이 증가하면서 선형적으로 학습 데이터를 100%까지 학습에 포함하는 방

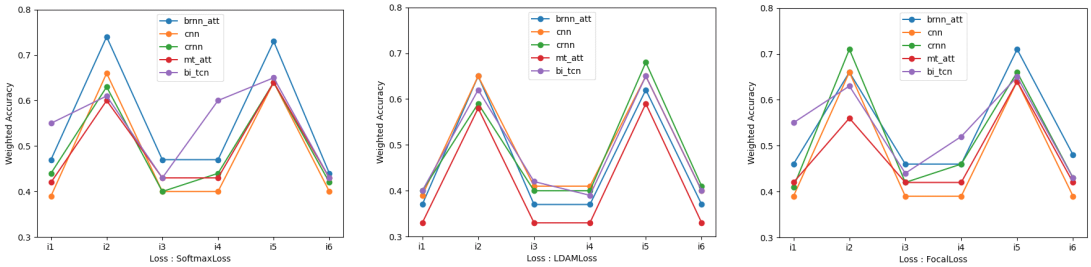

그림 14. 커리큘럼 학습에 대한 결과
Fig. 14. Results for Curriculum Learning

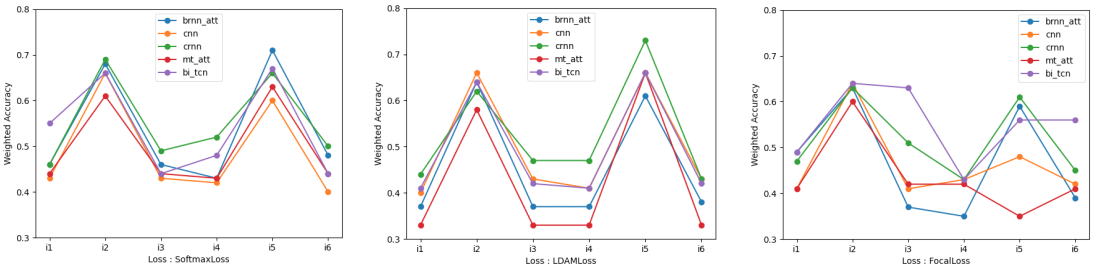

그림 15. 자기 페이스 학습에 대한 결과
Fig. 15. Results for self-paced learning

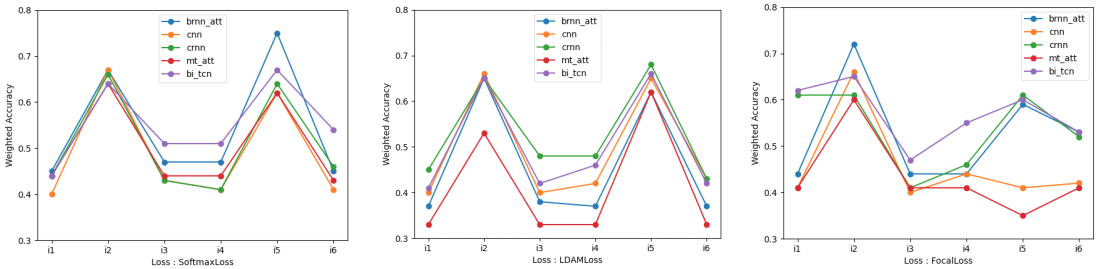

그림 16. 커리큘럼 및 자기 페이스 학습에 대한 결과
Fig. 16. Results for Curriculum and Self-Paced Learning

식을 사용하였다. 그림 15는 자기 페이스 학습에 대한 결과이다.

그림 16은 커리큘럼 학습 방법 및 자기 페이스 학습을 동시에 적용하였을 경우의 결과이다.

실험 결과를 보면 오버샘플링, Effective Number 방법을 사용하였을 경우 좋은 성능을 보였다. 표 2는 커리큘럼 학습 및 자기 페이스 학습 적용한 경우와 적용하지 않았을 경우를 비교한 것이다. Both는 커리큘럼 학습 방법 및 자기 페이스 학습을 동시에 적용을 의미한다. 표의 결과값은 네트워크 모델중 가장 좋은 WA 값을 나타낸다. 소프트맥스나 LDAM 손실함수의 경우에는 커리큘럼 학습 방법 및 자기 페이스 학습이 효과가 있으나 Focal 손실함수의 경우에는 오히려 커리큘럼 학습 방법 및 자기 페이스 학습이 성능을 낮추는 것을 볼 수 있다. 커리큘럼 학습 혹은 자기 페이스 학습이 항상 성능이 좋게 되는 것이 아니고, 손실함수에 따라 결과가 다름을 볼 수 있다.

표 2. 커리큘럼 학습 및 자기 페이스 학습 적용에 대한 결과
Table 2. Results for Curriculum Learning and Applying Self-Paced Learning

Imbalance	Method	Softmax	LDAM	Focal
Over Sampling	None	0.66	0.65	0.69
	Curriculum	0.74	0.65	0.71
	Self-Paced	0.69	0.66	0.64
	Both	0.67	0.66	0.72
Effective Number	None	0.7	0.68	0.76
	Curriculum	0.73	0.68	0.71
	Self-Paced	0.71	0.73	0.61
	Both	0.75	0.68	0.61

Ⅲ. 결론 및 향후 연구

본 논문에서는 데이터의 수가 적고, 클래스 불균형한 사이드미러 기어박스의 모터 소리 데이터를 이용한 불량 구분에 관한 다양한 실험을 진행하였다. 사용된 데이터는 일반적인 소리 데이터 클래스에서 사용되는 데이터와 다르게 대부분의 구간에서 비슷한 경향을 가지고 있으며, 일부 구간 혹은 주기적인 구간에서 다른 경향을 보이고, 실험을 통하여 다른 소리 데이터와 차이가 있음을 확인하였다. 사용한 데이터는 시계열 데이터이므로 다양한 시계열 처리가 가능한 네트워크 모델과 멜스펙트로그램을 통해 변환된 2차원 특징 벡터를 통한 이미지 방식의 네트워크 모델을 사용하였다. 데이터 증강 및 데이터 불균형의 문제를 개선하는 다양한 방법, 데이터의 난이도에 따른 학습을 하는 커리큘럼 학습 방법 및 자기 페이스 학습을 실험 및 비교하였다. 커리큘럼 학습 방법 및 자기 페이스 학습이 소프트맥스 손실함수에서 개선되는 것을 볼 수 있었다. 하지만 Focal 손실함수에서는 오히려 성능이 낮아지는 것을 볼 수 있었다. 가장 성능이 좋은 네트워크 모델은 Bidirectional LSTM Attention 이고, 손실함수는 Focal 손실함수, Effective Number 방식의 재가중치를 적용한 방법이었다. 하지만 네트워크 모델이 비교적 간단하고, 다양한 특징 벡터를 사용하지 않은 문제는 향후 연구를 통해 진행할 예정이다. 성능 개선을 위한 방법으로 부족한 데이터 및 클래스 불균형, 자기 지도학습을 통한 표현 학습등을 지속적으로 연구할 예정이다.

참 고 문 헌 (References)

[1] A. Khamparia, D. Gupta, N. G. Nguyen, A. Khanna, B. Pandey and P. Tiwari, "Sound Classification Using Convolutional Neural Network and Tensor Deep Stacking Network," in IEEE Access, vol. 7, pp. 7717-7727, 2019.
doi: http://doi.org/10.1109/ACCESS.2018.2888882

[2] K. Jaiswal and D. Kalpeshbhai Patel, "Sound Classification Using Convolutional Neural Networks," 2018 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), pp. 81-84, 2018.
doi: http://doi.org/10.1109/CCEM.2018.00021

[3] P. Tzirakis, J. Zhang and B. W. Schuller, "End-to-End Speech Emotion Recognition Using Deep Neural Networks," 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5089-5093, 2018.
doi: http://doi.org/10.1109/ICASSP.2018.8462677

[4] Zhichao Zhang, Shugong Xu, Tianhao Qiao, Shunqing Zhang, Shan Cao, "Attention Based Convolutional Recurrent Neural Network for Environmental Sound Classification", 2019. arXiv:1907.02230

[5] S. Wyatt et al., "Environmental Sound Classification with Tiny Transformers in Noisy Edge Environments," 2021 IEEE 7th World Forum on Internet of Things (WF-IoT), pp. 309-314, 2021.
doi: http://doi.org/10.1109/WF-IoT51360.2021.9596007

[6] Alexey Dosovitskiy, Lucas Beyer et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," 2020. arXiv:2010.11929

[7] Yue, Zhihan, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yu Tong and Bixiong Xu. "TS2Vec: Towards Universal Representation of Time Series.," 2021. arXiv:2106.10466

[8] Sepp Hochreiter, Jürgen Schmidhuber, "Long Short-Term Memory," Neural computation 9, 1735-80, 1997.
doi: http://doi.org/10.1162/neco.1997.9.8.1735

[9] B. Shi, X. Bai and C. Yao, "An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 11, pp. 2298-2304, 1 Nov. 2017.
doi: http://doi.org/10.1109/TPAMI.2016.2646371

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, "Attention is All you Need," 31st Conference on Neural Information Processing Systems (NIPS), 2017.

[11] Shaojie Bai, J. Zico Kolter, Vladlen Koltun, "An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling," 2018. arXiv:1803.01271

[12] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet classification with deep convolutional neural networks," Communications of the ACM, Volume 60, Issue 6, pp 84–90, June 2017.
doi: http://doi.org/10.1145/3065386

[13] Bengio, Y., Louradour, J., Collobert, R., & Weston, J. "Curriculum learning. In Proceedings of the 26th annual international conference on machine learning," pp. 41-48, June, 2009.

[14] Kumar, M., Packer, B., & Koller, D. "Self-paced learning for latent variable models," Advances in Neural Information Processing Systems 23 (NIPS 2010), pp. 1189-1197, 2010.

[15] Qingsong Wen, Liang Sun et al, "Time Series Data Augmentation for Deep Learning: A Survey," Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence Survey Track, pp. 4653-4660, 2021.
doi: http://doi.org/10.24963/ijcai.2021/631

[16] Park, Daniel S., et al. "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," Proc. Interspeech 2019, pp. 2613-2617, 2019.

[17] D. S. Park et al., "Specaugment on Large Scale Datasets," ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6879-6883, 2020.
doi: http://doi.org/10.1109/ICASSP40776.2020.9053205

[18] Xingcheng Song, Zhiyong Wu, Yiheng Huang, Dan Su, Helen Meng, "SpecSwap: A Simple Data Augmentation Method for End-to-End Speech Recognition," 2019. arXiv:1912.05533

[19] Helin Wang and Yuexian Zou and Wenwu Wang. "SpecAugment++: A Hidden Space Data Augmentation Method for Acoustic Scene Classification", arXiv:2103.16858v3

[20] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, Yannis Kalantidis "Decoupling Representation and Classifier for Long-Tailed Recognition," ICLR 2020.

[21] Y. Cui, M. Jia, T. -Y. Lin, Y. Song and S. Belongie, "Class-Balanced Loss Based on Effective Number of Samples," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9260-9269, 2019.
doi: http://doi.org/10.1109/CVPR.2019.00949

[22] T. -Y. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal Loss for Dense Object Detection," 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2999-3007, 2017.
doi: http://doi.org/10.1109/ICCV.2017.324

[23] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, Tengyu Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," Proceedings of the 33rd International Conference on Neural Information Processing Systems, Article No:140, pp.1567–1578, December 2019.

[24] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj and L. Song, "SphereFace: Deep Hypersphere Embedding for Face Recognition," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6738-6746, 2017.
doi: http://doi.org/10.1109/CVPR.2017.713

[25] H. Wang et al., "CosFace: Large Margin Cosine Loss for Deep Face Recognition," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5265-5274, 2018.
doi: http://doi.org/10.1109/CVPR.2018.00552

[26] J. Deng, J. Guo, N. Xue and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4685-4694, 2019.
doi: http://doi.org/10.1109/CVPR.2019.00482

[27] J. Salamon, C. Jacoby and J. P. Bello, "A Dataset and Taxonomy for

Urban Sound Research", 22nd ACM International Conference on Multimedia, Orlando USA, Nov. 2014.
[28] Livingstone SR, Russo FA (2018) The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. PLoS ONE 13(5): e0196391. doi: http://doi.org/10.1371/journal.pone.0196391
[29] Bosch, J. J., Janer, J., Fuhrmann, F., & Herrera, P. “A Comparison of Sound Segregation Techniques for Predominant Instrument Recognition in Musical Audio Signals”, in Proc. ISMIR, pp. 559-564, 2012.

저 자 소 개

장 일 식

- 2011년 2월 : 서울과학기술대학교 NID융합기술대학원 석사
- 2020년 3월 ~ 현재 : 서울과학기술대학교 지능형미디어연구센터 책임 연구원
- 2020년 9월 ~ 현재 : 서울과학기술대학교 나노IT디자인융합대학원 정보통신미디어공학전공 박사과정
- ORCID : https://orcid.org/0000-0003-0822-9857
- 주관심분야 : 컴퓨터비전, 딥러닝

박 구 만

- 1984년 2월 : 한국항공대학교 전자공학과 공학사
- 1986년 2월 : 연세대학교 전자공학과 공학석사
- 1991년 2월 : 연세대학교 전자공학과 공학박사
- 1991년 3월 ~ 1996년 9월 : 삼성전자 신호처리연구소 선임연구원
- 2016년 1월 ~ 2017년 12월 : 서울과학기술대학교 나노IT디자인융합대학원 원장
- 1999년 8월 ~ 현재 : 서울과학기술대학교 전자IT미디어공학과 교수
- 2006년 1월 ~ 2007년 8월 : Georgia Institute of Technology Dept.of Electrical and Computer Engineering, Visiting Scholar
- ORCID : https://orcid.org/0000-0002-7055-5568
- 주관심분야 : 컴퓨터비전, 지능형실감미디어