일반논문 (Regular Paper)
방송공학회논문지 제29권 제5호, 2024년 9월 (JBE Vol.29, No.5, September 2024)
https://doi.org/10.5909/JBE.2024.29.5.713
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

LCNN 기반 Deepfake Audio Detection의 개선 알고리즘

임 동 훈[a)], 한 종 기[a)‡]

Efficient Algorithms for Deepfake Audio Detection based on LCNN

Dong-Hoon Lim[a)] and Jong-Ki Han[a)‡]

요 약

오디오 딥페이크 관련 기술의 발전에 따라 기술의 악용이 현실화되고 있다. 딥보이스는 보이스피싱, 허위 정보 유포, 거짓 증거 등 다양하게 악용될 수 있으며, 이로 인해 발생할 사회적 피해 또한 상당하다. 그러나 딥보이스 탐지 기술은 여러 가지 한계를 가지고 있으며, 실제로 적용하기 어려운 기술이다. 구체적인 이유는 다음과 같다. 첫째, 이미지에 비해 음성은 언어, 주변 잡음 등 성능에 직접적으로 영향받는 요소가 많다. 둘째, 모델의 일반화 성능개선과 실시간 탐지를 위한 경량화는 상충 관계(trade-off)가 있다. 셋째, 학습 데이터에 대해 의존적이며, 학습하지 않은 언어나 새로운 딥보이스 시스템에 대해서는 모델의 탐지 성능이 매우 하락한다. 기존 모델들도 일반적으로 특정 데이터셋에 대해서만 성능이 준수하거나, 일반화 성능이 부족한 경우가 많다. 본 논문에서는 이를 개선하기 위해 LCNN-LSTM 모델을 백본으로 선택한 후, 세 가지 세부 기술들을 제안하고, 이를 구현해 기존의 몇 가지 모델과 비교하여 일반화 성능을 평가한다. 특히 주변 잡음이나 새로운 언어(한국어)에 대한 성능을 실험하여 앞으로의 딥보이스 탐지 기술의 발전이 나아갈 방향에 대해 모색한다.

Abstract

The rapid development of audio deepfake technology has facilitated its use in various malicious activities, including voice phishing, the dissemination of misinformation, and the fabrication of evidence, all of which pose significant societal risks. However, the detection of deepfake voices presents substantial challenges. First, unlike visual media, audio is significantly influenced by factors such as language and background noise. Second, there is a trade-off between enhancing the generalization performance of detection models and optimizing them for lightweight, real-time applications. Third, these models often exhibit strong dependency on their training data, resulting in a notable decline in detection performance when encountering unfamiliar languages or novel deepfake systems not represented in the training set. Many existing models perform adequately only on specific datasets and frequently demonstrate limited generalization capabilities. To address these challenges, this paper proposes three enhancements to the LCNN-LSTM architecture and evaluates their effectiveness through comparative analysis with several established models. Specifically, the study assesses the performance of these models in noisy environments and their adaptability to new languages, including Korean, to identify potential directions for the advancement of deepfake voice detection technology.

Keyword : Audio Deepfake, Deep learning, Deepvoice, Anti-spoofing

Ⅰ. 서 론

딥페이크 오디오(Deepfake Audio) 혹은 딥보이스(Deepvoice)는 딥페이크(Deepfake)와 비슷하지만, 이미지가 아닌 음성을 생성한다는 점에서 다르다. 딥페이크 오디오(이하 딥보이스라고 칭함)는 일반적으로 1) 텍스트-음성 변환(Text-to-Speech, TTS), 2) 음성 변환(Voice Conversion, VC), 3) 감정 변조(Emotion Fake), 4) 장면 변조(Scene Fake), 5) 부분 변조(Partially Fake)로 나눌 수 있다[1].

먼저 텍스트-음성 변환은 주어진 텍스트를 자연스러운 음성으로 생성하는 것이 목적이다. 이는 전통적인 음성 조각 결합 방식(Concatenative TTS) 및 엔드 투 엔드 딥러닝 방식(End-to-End TTS)을 포함한다. 음성 합성 시스템 전체는 음소 토큰 생성부터 보코더(vocoder)까지 복잡한 구조로 이루어진다. Tacotron[2], WaveNet[3], Waveglow[4], hifiGAN[5] 등의 활용 가능한 모델이 있으며 최근에는 하나의 모델이 여러 대상 목소리를 출력하는 제로샷(zero-shot) TTS(YourTTS[6], XTTS[7] 등)도 공개되었다. 음성 변환은 주어진 음성을 다른 화자의 목소리로 변환하는 것이 목적이다. StarGAN-VC[8], StreamVC[9] 등의 모델이 있다. 감정 변조는 음성의 감정을 변조하여 원래 감정과 다른 감정을 표현하는 것으로 이미지의 스타일 변환과 유사하다. VoiceGAN[10]이 대표적이다. 장면 변조는 음성의 배경 소리를 변경하여 다른 장면으로 변조하는 것을 말한다. SEGAN[11]이 대표적이다. 부분 변조는 음성의 특정 부분이나 단어만 변환하는 기술로, WaveNet을 활용한 오디오 인페인팅(audio-inpainting) 등을 통해 가능하다[12].

이런 다양한 시스템을 통해 생성된 가짜 음성은 허위 정보 유포, 지인 사칭과 같은 고도화된 보이스피싱, 법적 허위 증거 등 다양하게 악용될 수 있으며, 이로 인해 발생할 것으로 예상되는 사회적 피해 또한 상당하다. 딥보이스 탐지는 이러한 가짜 음성을 식별하고 차단하기 위한 기술로, 인공지능과 머신러닝 알고리즘을 활용해 오디오의 진위를 파악하는 데 주목적이 있다. 누구나 오픈소스를 통해 손쉽게 고품질의 딥보이스를 생성할 수 있는 시대가 된 만큼 탐지 기술 연구의 필요성도 증가하고 있다.

이런 다양한 딥보이스를 식별하기 위해서 여러 분류기(Classification Algorithms)가 연구되었다. 전통적인 머신러닝 방식으로는 서포트 벡터 머신(SVM)과 가우시안 혼합 모델(GMM)등이 있지만 한계가 존재하며[1], 최근에는 인공지능을 활용해 딥보이스와 실제 오디오의 차이(비정상적인 특징)에 대해 감지하는 접근 방식이 대부분이다. 널리 사용되는 대표적인 몇 가지 연구의 요약은 다음과 같다.

Zhenzong Wu[13]는 딥페이크 탐지 기술에 활용되던 CNN(Convolutional Neural Network)구조 기반의 Light CNN(LCNN)을 딥보이스 탐지에 활용하는 방식을 제안했다. LCNN은 CNN 레이어에 Max-Feature-Map(MFM)을 추가하여 특징 채널을 두 그룹으로 나누고, 요소별 최대 값만 취해 유효한 특징을 추출하도록 유도하여 성능을 높인다. Piotr Kawa[15]는 스펙트로그램 기반으로 ResNet[14]에 채널 어텐션(Channel attention)을 적용해 성능을 개선한 SpecRNet을 제안했다. 블록을 짧게 구성하여 가볍고 빠르면서도 LCNN과 비슷한 성능을 달성하여 실시간 탐지에 적합하다. Darius Afchar[16]는 딥페이크 비디오 탐지를 위해 이미지에서 중간 규모의 특징(Mesoscopic properties)을 강조하여 추출하는 MesoNet과 MesoInception-4를 제안한다. 이는 비디오 탐지에서 좋은 성능을 보였으며, 딥보이스 탐지에서도 활용할 수 있다. Piotr Kawa[18]는 음성인식 모델인 Whisper[17]을 전처리 frontend로 사용한 개선 방식을 제안했다. 세 가지 모델 LCNN[13], SpecRNet[15], MesoNet[16]에 대해 적용하면 성능이 개선되어 ASVspoof 2021 DF 데이터셋[19]에서 평균 21%의 EER(Equal Error Rate) 감소를 얻었다.

위와 같은 여러 연구가 수행되었음에도, 딥보이스 탐지 분야에는 여전히 해결해야 할 문제가 많은데, 그 이유는 다음과 같다. 첫째, 음성은 이미지와 달리 언어마다, 지역마다, 사람마다 그 억양과 말투가 달라지므로 언어에 따라 탐지 성능이 크게 달라진다. 둘째, 녹음 환경에 따라 그 품질

a) 세종대학교 전자정보통신공학부(Electrical Engineering, Sejong University)
‡ Corresponding Author : 한종기(Jong-Ki Han)
E-mail: hjk@sejong.edu
Tel: +82-2-3408-3739
ORCID: https://orcid.org/0000-0002-5036-7199
※ This work was supported by the National Research Foundation of Korea (NRF) under Grant 2022R1F1A1071513 funded by the Korea government through the Ministry of Science and ICT (MSIT).
· Manuscript August 13, 2024; Revised August 19, 2024; Accepted August 19, 2024.

이나 주변 잡음, 오디오 포맷, 압축 정도가 달라지므로 성능에 영향이 있다. 셋째, 딥보이스 탐지 기술은 새로운 유형의 공격이나 우회 공격에 성능이 저하되기 쉽다. 넷째, 실시간 탐지, 모바일 기기나 임베디드 시스템에서의 적용을 위해 탐지 시스템이 가볍고, 빠르고, 정확해야 한다. 이런 특징으로 인해 딥보이스 탐지 모델은 적은 데이터를 통해 일반화하는 능력이 매우 중요하다.

한편 딥페이크 이미지 탐지에 관한 연구는 많이 이루어지고 있지만, 음성에 관한 탐지 연구는 아직 미비하다. 딥페이크 관련 대회(FaceForensics++, DFDC, Deeper Forensics Challenge, DFGC, OpenMFC, ForgeryNet 등)가 다양하게 개최되는 것과 달리, 딥페이크 음성 탐지 분야는 상대적으로 대회(ASVspoof, ADD 등)가 부족하며, 공개된 데이터셋 또한 많지 않다. 이런 문제는 위의 연구들에서도 나타난다. 대부분의 실험이 영어 및 중국어 데이터셋에서 이루어지며, 데이터셋에 포함된 음성 합성 시스템도 많지 않다.

본 논문에서는 먼저 LCNN-LSTM 모델을 중심으로 일반화 성능을 높이기 위한 몇 가지 개선을 제안하고자 한다. 본 논문에서는 음성 외 신호 강조를 위한 고주파 필터를 추가했고, 특징 압축 방법을 수정했으며, 특징 강조 블록을 추가했다. 또한 기존에 적은 언어에 대해서만 제안된 모델들을 다국어 데이터셋에서 학습하여 성능을 평가하고, 실제 환경의 오디오와 같이 주변 잡음이 추가된 데이터에 대해서도 평가를 진행하여 일반화 성능을 평가하고자 한다. 나아가 다국어 데이터에 한국어가 포함되어 있지 않으므로, 한국어만으로 이루어진 소규모 데이터에 대해 테스트하여 한국어에 대한 성능을 평가해 보고자 한다.

본 논문의 구성은 다음과 같다. Ⅱ장에서는 기존의 딥보이스 탐지에 관한 연구를 소개하고 연구의 방향을 설명한다. Ⅲ장에서는 LCNN-LSTM을 개선하는 몇 가지 구조를 제안한다. Ⅳ장에서는 실험 환경과 사용 데이터를 설명하고, 실험 결과를 기반으로 기존 기술과 제안기술을 비교 및 분석한다. Ⅴ장에서는 결론을 내린다.

Ⅱ. 기존 연구

1. LCNN-LSTM

LCNN[13]은 일반적으로 딥보이스 탐지에서 양방향 LSTM(BiLSTM)과 함께 사용된다. STC(Speech Technology Center)팀은 LCNN 모델을 활용해 ASVspoof 2019에서 SOTA(State Of The Art)를 달성했으며, 이후 ASVspoof 2021에서도 LCNN-LSTM이 기준 모델(Baseline model)로 제시되었다[19][20][21].

백본 모델인 LCNN-LSTM의 전체적인 구조는 그림 1과 같으며, 구체적인 구조는 표 1과 같다. 먼저 front에서 원본 오디오의 전처리(MFCC)를 수행한 후, LCNN 블록의 입력으로 사용한다. 입력 텐서는 '배치 크기', '채널', '특징 수', '프레임 수'를 표현하는 정보들의 결합으로 구성된다. 입력이 일정해야 하므로 샘플을 일정 길이로 자르거나 패딩을 진행한다. LCNN 블록에서는 Conv2D 및 MFM

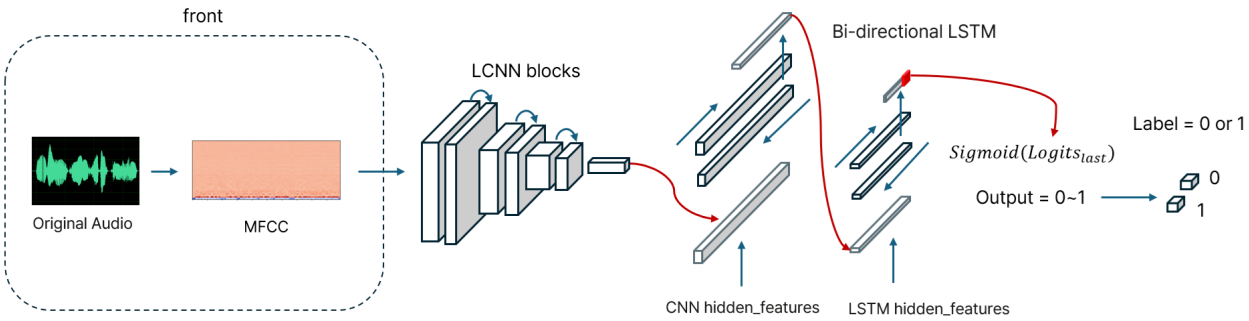

그림 1. LCNN-LSTM 모델 구조
Fig. 1. Architecture of the LCNN-LSTM model

(MaxFeatureMap2D)를 여러 차례(예를 들면, 10회) 수행하여 특징을 추출한다. MFM 레이어에서는 특징을 채널 차원으로 두 그룹으로 나눈 뒤, 요소별 최대값(Elelemt-wise Maximum)을 취해 특징 차원을 절반으로 줄인다. 이어서 추출한 features를 양방향 LSTM(Bidirectional LSTM)을 사용해 분류한다. 두 개의 BiLSTM layer는 복잡한 패턴이나 장기 인식도 가능하도록 해 주며, 시계열 데이터 분류에 좋은 성능을 보인다. 마지막으로 출력의 마지막 logit 값의 sigmoid를 통해 결과를 예측한다. 이후 이를 실제 정답과 비교해 이진 크로스 엔트로피(Binary Cross Entropy)를 계산하고, 학습을 진행한다.

표 1. LCNN-LSTM 구조
Table 1. Table of LCNN-LSTM Structure

Layer Type	Parameters
Conv2d	Input Channels: 1, Output Channels: 64
	Kernel Size: (5, 5)
	Stride: (1, 1), Padding: (2, 2)
MaxFeatureMap2D	Max Dim: 1
MaxPool2d	Kernel Size: (2, 2), Stride: (2, 2)
Conv2d	Input Channels: 32, Output Channels: 64
	Kernel Size: (1, 1), Padding: (0, 0)
MaxFeatureMap2D	Max Dim: 1
BatchNorm2d	Num Features: 32
Conv2d	Input Channels: 32, Output Channels: 96
	Kernel Size: (3, 3), Padding: (1, 1)
MaxFeatureMap2D	Max Dim: 1
MaxPool2d	Kernel Size: (2, 2), Stride: (2, 2)
BatchNorm2d	Num Features: 48
Conv2d	Input Channels: 48, Output Channels: 96
	Kernel Size: (1, 1), Padding: (0, 0)
MaxFeatureMap2D	Max Dim: 1
BatchNorm2d	Num Features: 48
Conv2d	Input Channels: 48, Output Channels: 128
	Kernel Size: (3, 3), Padding: (1, 1)
MaxFeatureMap2D	Max Dim: 1
MaxPool2d	Kernel Size: (2, 2), Stride: (2, 2)
Conv2d	Input Channels: 64, Output Channels: 128
	Kernel Size: (1, 1), Padding: (0, 0)
MaxFeatureMap2D	Max Dim: 1
BatchNorm2d	Num Features: 64
Conv2d	Input Channels: 64, Output Channels: 64
	Kernel Size: (3, 3), Padding: (1, 1)
MaxFeatureMap2D	Max Dim: 1
MaxPool2d	Kernel Size: (2, 2), Stride: (2, 2)
Conv2d	Input Channels: 32, Output Channels: 64
	Kernel Size: (1, 1), Padding: (0, 0)
MaxFeatureMap2D	Max Dim: 1
BatchNorm2d	Num Features: 32
Conv2d	Input Channels: 32, Output Channels: 64
	Kernel Size: (3, 3), Padding: (0, 0)
MaxFeatureMap2D	Max Dim: 1
MaxPool2d	Kernel Size: (2, 2), Stride: (2, 2)
Dropout	Dropout Rate: 0.7
BLSTMLayer	BLSTMLayer(512, 512)
BLSTMLayer	BLSTMLayer(512, 512)
Linear	Input Features: 512, Output Features: 1

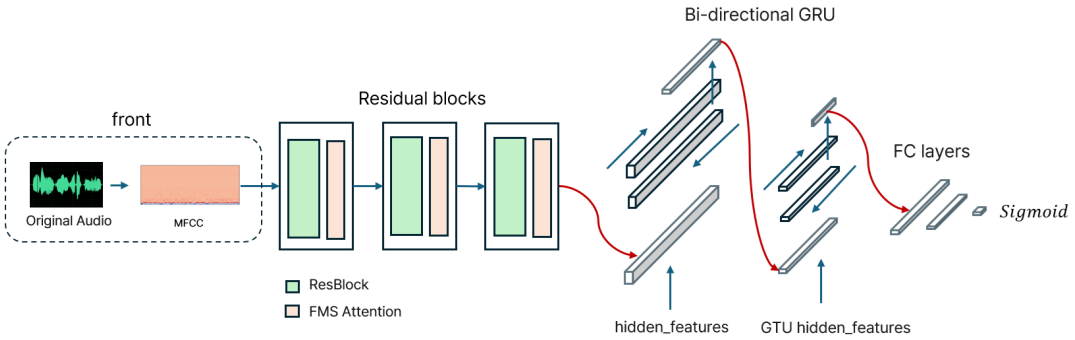

그림 2. SpecRNet 모델 구조[15]
Fig. 2. Architecture of the SpecRNet model[15]

2. SpecRNet

SpecRNet의 구조는 그림 2와 같다[15]. LCNN-LSTM과 마찬가지로 원본 오디오를 MFCC로 전처리 후 잔차 블록(Residual block) 및 FMS 어텐션을 3번 통과하여 특징을 추출한 후, 이를 양방향 GRU 및 완전 연결 층을 통해 가짜와 진짜 오디오를 분류하게 된다.

잔차 블록의 구체적인 구조는 그림 3과 같다. 배치 정규화와 컨볼루션 레이어를 두 번 적용하며, 잔차 연결에도 컨볼루션 레이어를 적용한다. 이후 FMS 어텐션 블록에서는 최대 풀링을 적용하며 그 과정에서 전역 평균 풀링 및 선형 연결층을 통해 어텐션을 구현한다.

SpecRNet은 빠르면서도 LCNN과 비슷한 성능을 보였지만 모델의 깊이가 얕아 일반화 성능이 다소 부족한 문제가 있다.

3. MesoInception4

MesoInception4은 MseoNet4에 인셉션 블록[22]을 사용한 모델이다. 전체 구조는 그림 4와 같으며, 인셉션 블록의 구조는 그림 5와 같다. 조작된 얼굴 영상을 탐지하는 모델이지만 딥보이스 탐지에도 활용할 수 있다. 이 기술 또한 특징 의존적이며, 실시간 탐지를 위해 모델이 매우 경량화되어 있다. 따라서, 이 기술은 일반화 성능이 부족한 문제가 존재한다.

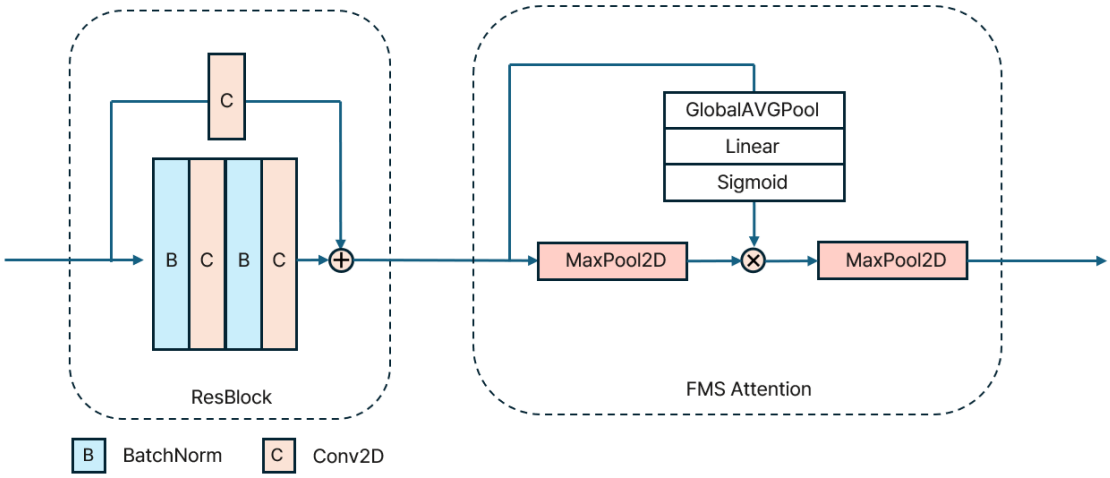

그림 3. SpecRNet 내부 잔차 블록 구조
Fig. 3. Architecture of Residual Block in SpecRNet model

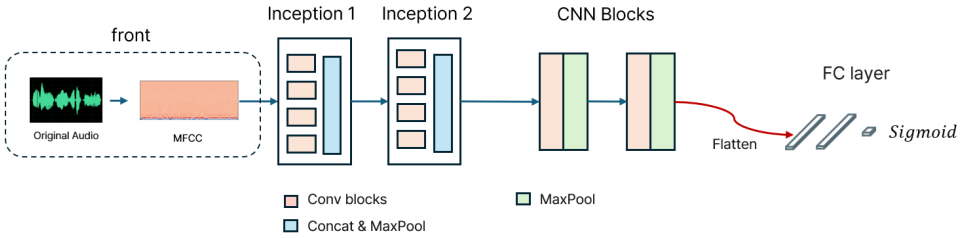

그림 4. MesoInception4 모델 구조[16]
Fig. 4. Architecture of the MesoInception4 model[16]

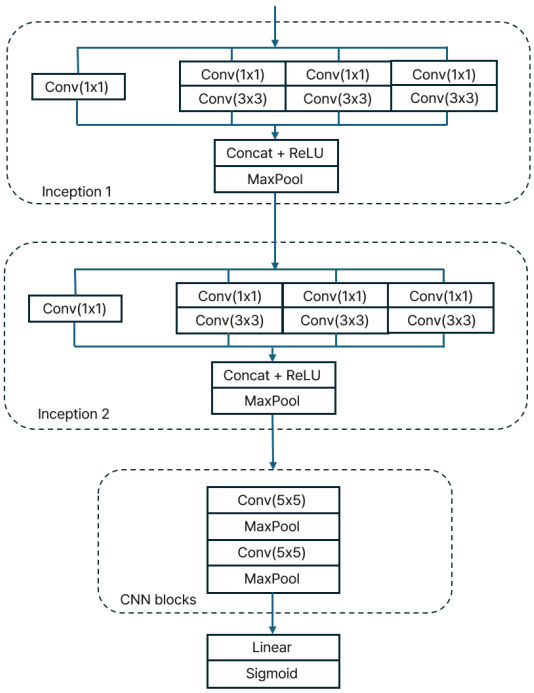

그림 5. 인셉션 블록 및 컨볼루션 블록 구조[16]
Fig. 5. Architecture of the Inception block and CNN block

4. Whisper-LCNN

Whisper-LCNN은 전처리 과정에 음성인식 모델인 whisper-tiny.en을 활용한 모델이다[18]. Whisper는 트랜스포머(Transformer) 기반 인코더-디코더 모델로, 자동 음성 인식(ASR) 및 음성 번역을 위한 사전 훈련된 모델이다[17]. 사전 훈련된 모델은 대량의 데이터를 통해 학습되어 미세 조정 없이도 많은 데이터 세트와 도메인으로 일반화할 수 있다. 그 중, 영어에 대해서만 학습된 whisper_tiny.en을 사용해 LCNN-LSTM의 전처리부(front)로 사용한다. 제안된 논문에서도 영어만 있는 테스트 데이터에서 개선되었으므로 다른 언어에 대한 성능 평가가 필요하다[18].

Ⅲ. 제안하는 시스템 및 알고리즘

1. 제안하는 시스템

LCNN-LSTM의 성능을 개선하기 위해 다음과 같이 4가지 기술을 제안한다. 그림 7에서 (A) HPF(High Pass Filter,

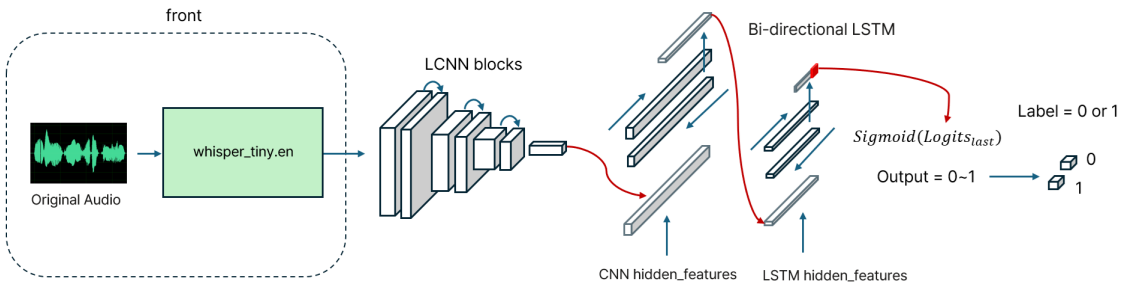

그림 6. Whisper를 전처리로 사용한 LCNN 구조
Fig. 6. Architecture of the Whisper-LCNN

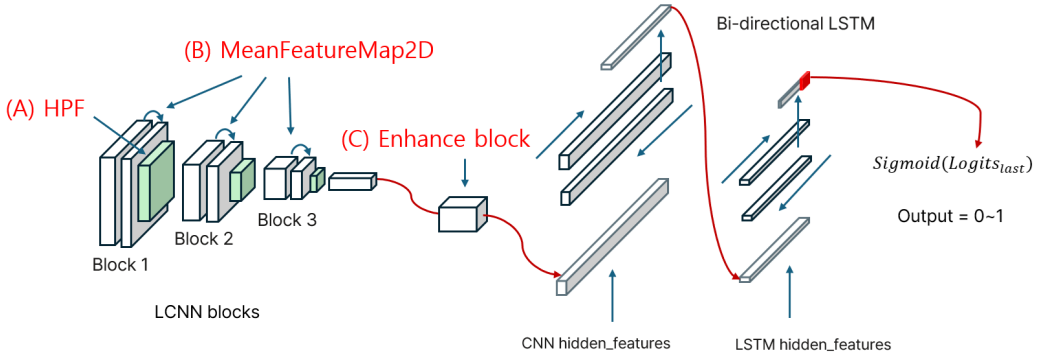

그림 7. LCNN-LSTM 모델 구조[14]
Fig. 7. Architecture of the LCNN-LSTM model[14]

하이패스 블록)은 고주파 영역에 대한 특징을 강화하고, (B) MeanFeatureMap2D(평균 특징 맵)은 모델이 채널 차원의 압축을 더 부드럽게 수행하도록 한다. 그리고 (C) Enhance Block(특징 강조 블록)를 통해 LSTM의 입력을 개선한다. 마지막으로 (D) 데이터 증강 과정을 통해 더 일반적인 상황에서의 테스트를 진행하도록 한다.

2. HPF(High Pass Filter)

인간의 목소리는 50~4000Hz 내의 주파수 영역에서 에너지가 가장 크다. 음성을 이해하기 위해서는 해당 주파수 범위 내의 정보만 중요하다. 즉 사람이 딥보이스를 진짜와 같이 느끼기 위해 중요한 주파수 영역도 4000Hz 이하라는 것이다. 생성 모델 또한 해당 주파수 영역을 더 높은 품질로 생성하도록 학습하게 된다. 하지만 딥보이스 탐지에서는 오히려 그 외의 고주파 영역의 특징으로부터 가짜 오디오임을 탐지하는 것이 유리할 수 있다. 이미지 생성 모델에서 피사체와 주변의 흐름이나 배경의 품질 차이가 나타난다는 연구가 있는 것처럼, 음성도 음성 영역과 그 외 영역에서의 미세한 왜곡이나 패턴이 나타날 수 있으며, 이러한 부자연스러운 소리나 왜곡의 탐지가 성능을 개선할 수 있을 것이라 가정했다[23]. 따라서 LCNN이 고주파 영역에 더 집중하도록 개선해 탐지 성능 차이가 생기는지 실험해 보고자 한다.

이를 위해 그림 8과 같이 주파수 축에 대해서만 선형으로 0.5부터 1까지의 값을 가지는 텐서(하이패스 윈도우)를 입력 특징 맵과 같은 크기로 생성하고, 입력과 요소별 곱셈을 통해 고주파 영역을 강조하는 하이패스 블록을 구성했다. 그림 8에서 밝은색이 더 큰 값을 나타낸다.

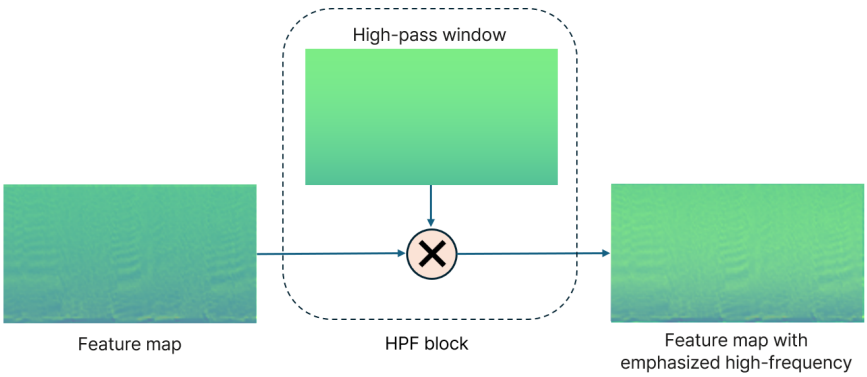

그림 8. 하이패스(HPF) 블록 구조
Fig. 8. Architecture of the high-pass block

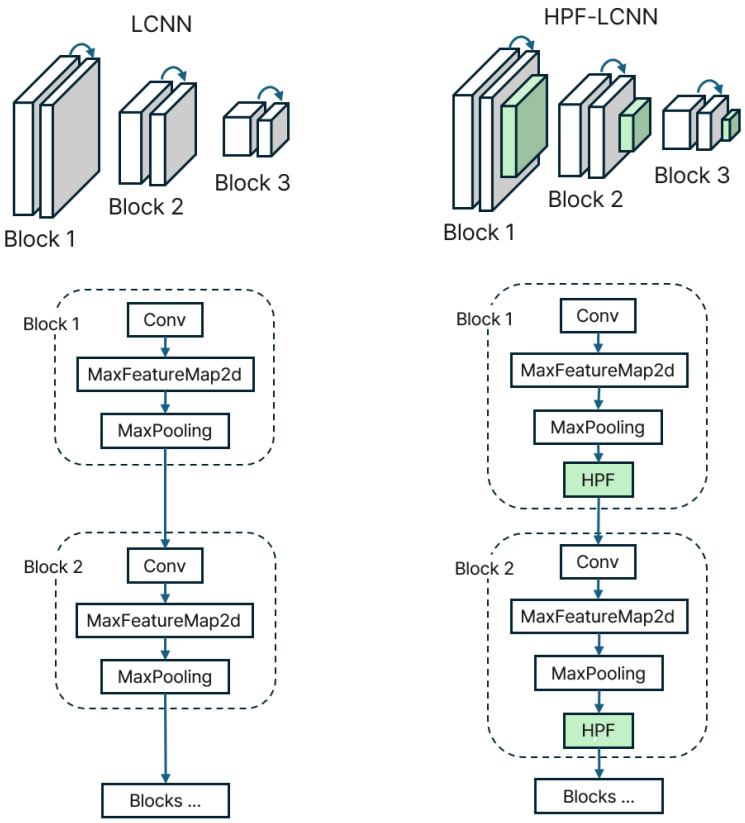

그림 9. 기존 LCNN과 하이패스 블록이 추가된 LCNN 구조
Fig. 9. Architecture of the LCNN with HPF block

그림 9는 하이패스 블록이 추가된 LCNN의 구조를 보여준다. 최대 풀링 이후 고주파 성분 강조를 수행한다.

3. MeanFeatureMap2D(평균 특징 맵)

기존 LCNN의 최대 특징 맵(MaxFeatureMap2D)을 평균 특징 맵(MeanFeatureMap2D)으로 대체한다.

기존 최대 특징 맵은 채널 차원을 기준으로 특징을 두 그룹으로 나누어, 요소별 최대 값(Element-wise Max)을 추출하여 채널 차원을 절반으로 줄인다. 그림 10과 같이 여러 커널 묶음으로부터 얻은 값 중 더 큰 값이 선택된다. 이는 커널마다 OR 연산을 하는 것과도 같다. 이 과정에서 정보

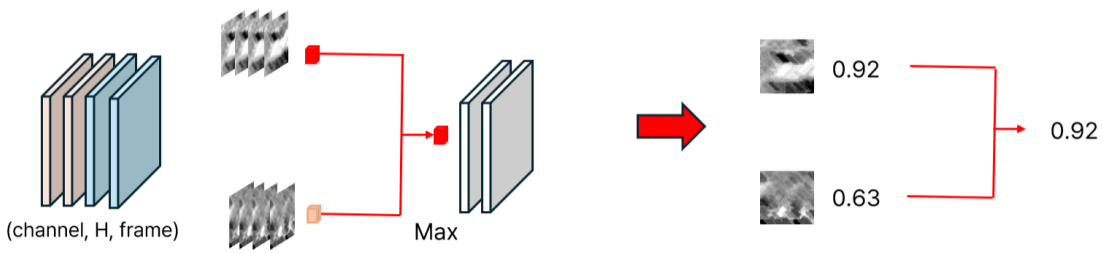

그림 10. 기존 최대 특징 맵의 요소 계산 예시
Fig. 10. Example of Calculating Elements in a MaxFeatureMap

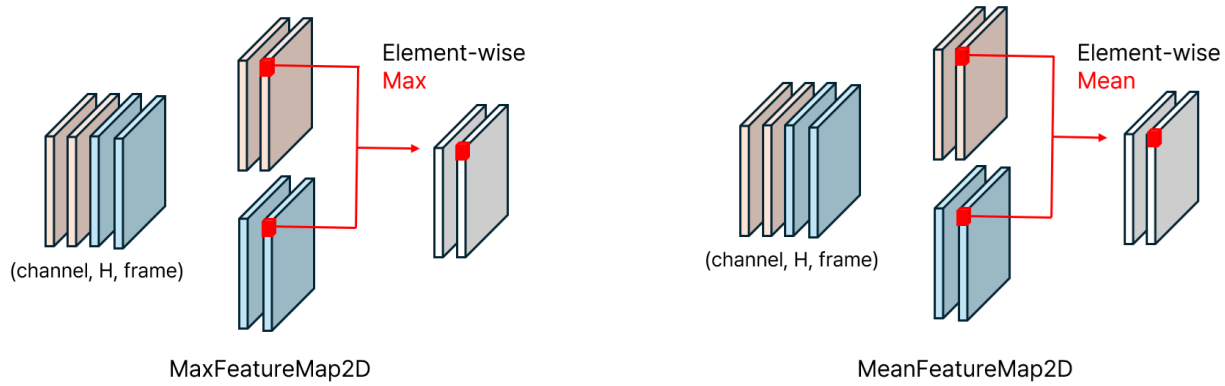

그림 11. 최대 특징 맵과 평균 특징 맵 흐름도
Fig. 11. Flow of MaxFeatureMap and MeanFeatureMap

의 압축과 같은 효과가 학습을 빠르고 효과적으로 이루어지도록 하지만, 특정 커널의 결과 정보는 손실된다.

이 과정을 평균값 계산 과정으로 대체하면, 두 특징의 중간을 얻어 정보의 손실을 방지하고, 더 부드럽게 학습할 수 있을 것으로 예상한다. 기존 구조는 그림 11의 왼쪽과 같고, 개선된 구조는 그림 11의 오른쪽과 같다.

4. Enhance Block(특징 강조 블록)

LSTM의 입력에 특징 강조 블록을 추가한다.

제시하는 특징 강조 블록(Enhance block)은 그림 12와 같다. 먼저 입력 특징에 소프트맥스를 적용하여 0~1 사이의 확률값으로 변환한다.

$$prob = Softmax(x) \quad (\dim = 2) \tag{1}$$

그림 12에는 시각화를 위해 랜덤 2차원 데이터에 대한 예시를 보이고 있지만, 실제 입력 특징 X는 '배치 크기', '채널 수', '특징 수'를 결합한 형태로 구성되어 있으므로, 각 채널에 대해 1차원 데이터를 같은 과정으로 변환해 prob를 얻는다.

$$d = 1 - prob * \log(prob) \tag{2}$$

이후 식 (1)에서 얻은 prob에 (2)와 같이 prob에 대해 요소별 계산을 통해 d를 계산한다. [0, 1] 구간 사이의 d의 그래프는 그림 13과 같이 위로 볼록한 포물선과 비슷하다. 이

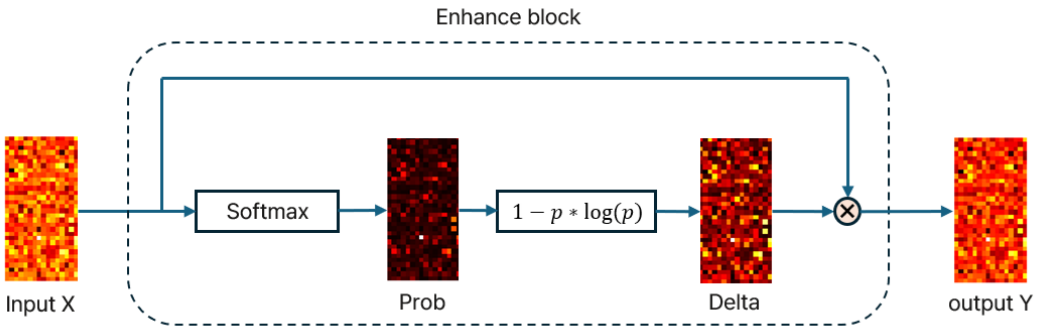

그림 12. 특징 강조 블록 구조
Fig. 12. Architecture of the Enhance block

는 너무 값이 급격하게 작아지지 않도록 보정하는 역할을 한다.

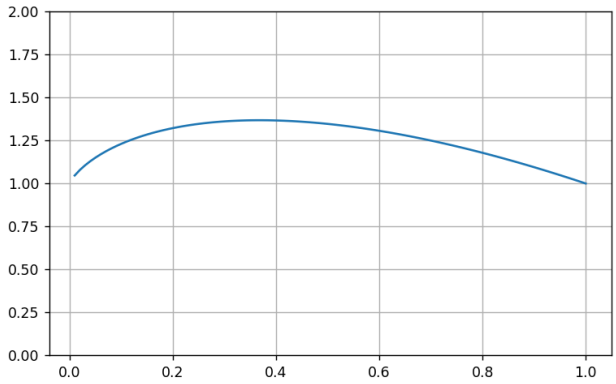

그림 13. 식 (2)의 그래프
Fig. 13. Graph of Equation (2)

$$y = x * d \tag{3}$$

마지막으로 출력 y는 입력 x와 d를 곱하여 식 (3)과 같이 얻어진다.

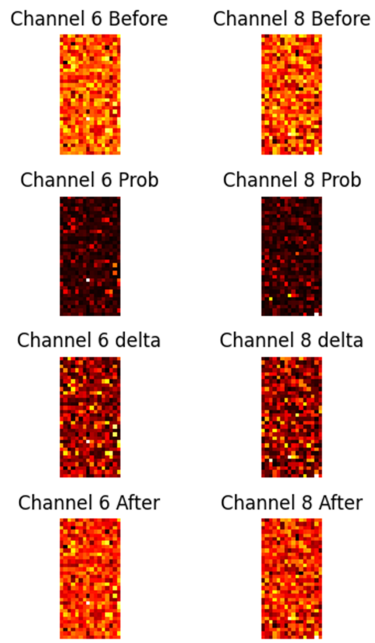

그림 14. 중간 단계의 특징 변화 예시
Fig. 14. Example of Feature Changes at Each Stage

위 과정을 통해 큰 값은 그대로, 작은 값은 작을수록 더 작아지게 하여 중요한 특징만 집중하도록 하고자 했다. 그림 14에 변화 과정 예시를 나타내었다.

5. 데이터 증강

실험용 데이터와 달리 실제 환경에서는 다양한 잡음과 주변 소음이 포함되는 경우가 대부분이다. 따라서 모델의 강건성과 일반화 능력을 테스트하기 위해 데이터 증강을 진행한다. 데이터 증강 과정은 그림 15와 같다.

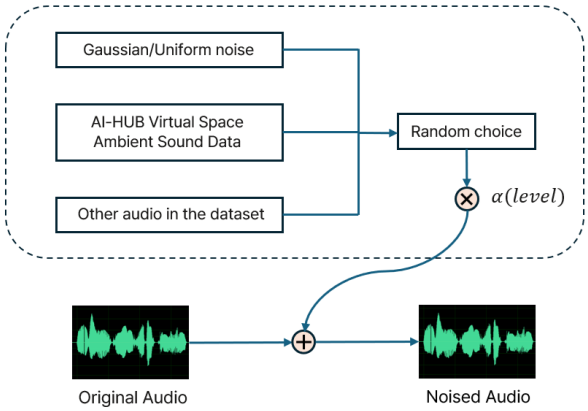

그림 15. 데이터 증강 과정 흐름도
Fig. 15. Flowchart of Data Augmentation Process

먼저 가우시안 노이즈(Gaussian noise) 또는 균일 노이즈(Uniform noise)를 추가하는 방식, AI-HUB의 가상공간 환경음 매칭 데이터[24]를 추가하는 방식, 데이터셋 내의 다른 오디오를 추가하는 방식 중 랜덤하게 한 개를 선택한다. 균일 노이즈의 경우 -1에서 1사이의 값을, 가우시안 노이즈의 경우 평균이 0이고 표준편차가 1인 잡음을 생성한다. 이후 추가되는 오디오의 크기를 조절하기 위해 상수 α를 곱하여 신호의 크기를 줄인다. 이를 원본 오디오와 같은 크기로 패딩 및 자르기를 수행한 뒤 원본 오디오에 더해준다. 본 논문에서는 크기 상수 α를 0.001로 설정했다.

이를 통해 카페나 길거리 등 특정 환경에서의 음성, 노이즈가 추가된 음성, 다중화자 음성 등 더 일반적인 상황에 대한 오디오를 생성할 수 있다. 해당 과정은 훈련 시에도 활용될 수 있지만, 본 논문에서는 테스트 과정에서 잡음이

추가된 데이터를 활용한 실험으로 모델의 강건성을 테스트하고자 한다.

Ⅳ. 실험 결과 및 분석

1. 실험 조건

본 논문에서 제안한 기술들의 성능을 평가하기 위해서, 기존에 공개된 LCNN-LSTM[25] 시스템에 (A) 하이패스 블록, (B) 평균 특징 맵, (C) 특징 강조 블록 세 가지 기술들을 구현하였다. 본 논문의 제안 기술들이 반영된 시스템의 성능을 기존 기술들인 LCNN-LSTM, Whisper-LCNN, Spec-RNet, MesoNet의 성능들과 비교하였다. 전처리 MFCC 단계는 Window 길이는 400, Hop 길이는 160으로 설정했으며, mfcc 계수는 128개, n_fft는 512로 설정했다.

제안하는 시스템 및 기존 시스템들을 학습시키기 위해 다국어 딥보이스 데이터셋인 MLAAD[26][27]와 실제 음성 낭독 데이터셋인 M-AILABS speech dataset[28]를 사용하였다. MLAAD는 독일의 IT 보안 연구 기관인 Fraunhofer AISEC에서 2024년 4월에 공개한 데이터셋으로, 23개 언어에 대해 총 53개의 시스템을 통해 생성한 딥보이스 데이터셋이다. 한국어는 포함되어 있지 않다. 자세한 정보는 그림 16과 그림 17에 나타냈다[26]. 그림에서 확인할 수 있듯이,

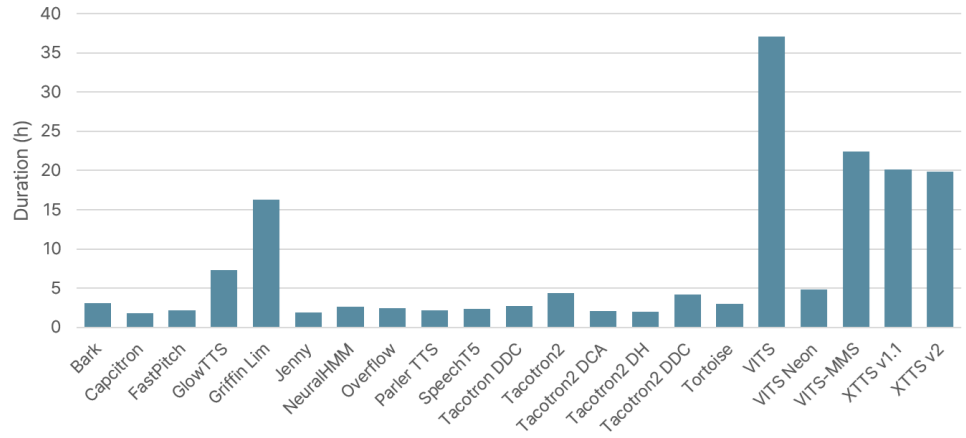

그림 16. MLAAD 내의 시스템별 시간
Fig. 16. Duration per System within MLAAD

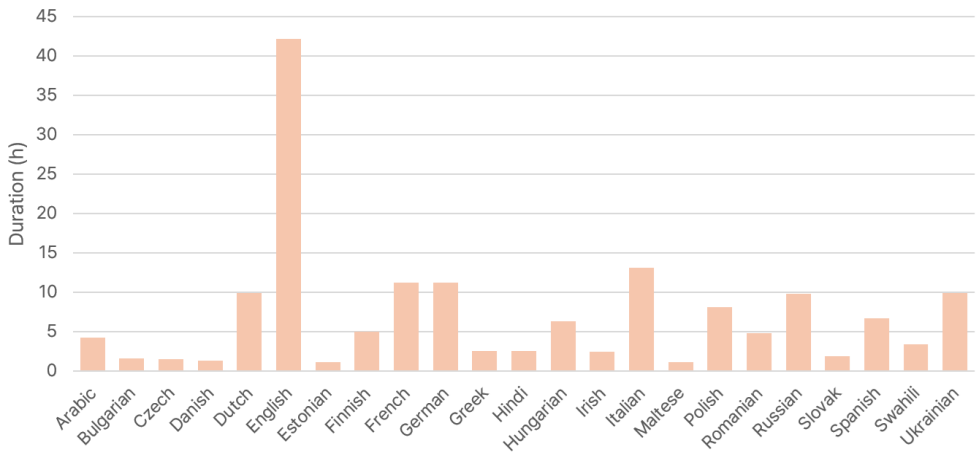

그림 17. MLAAD 내의 언어별 시간
Fig. 17. Duration per Language within MLAAD

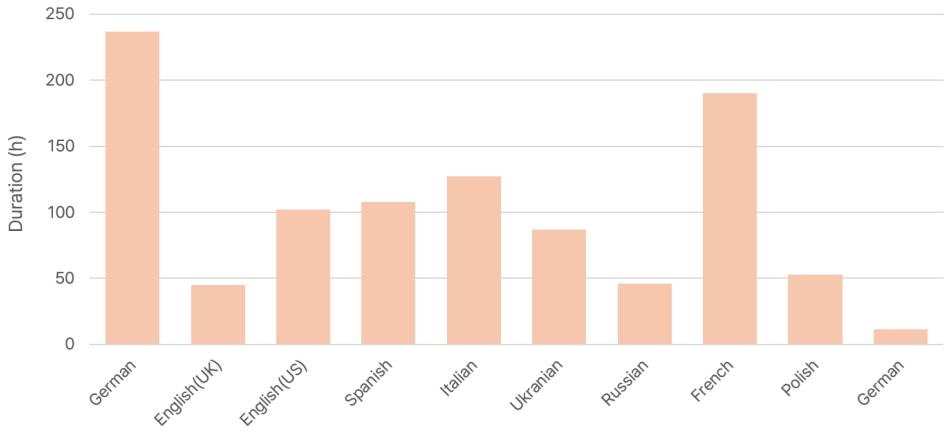

그림 18. M-AILABS 내의 언어별 시간
Fig. 18. Duration per Language within M-AILABS

이 데이터에서 영어의 비중이 가장 크다. MLAAD는 총 76,000개의 음성 WAV 파일로 구성된다. 각각 한 문장 단위로 생성된 WAV 파일이며, 샘플링 주파수는 16,000Hz이다. MLAAD는 가짜 오디오만 있으므로, 실제 음성인 M-AILABS와 함께 사용해야 한다. M-AILABS speech dataset(이하 M-AILABS)는 음성인식 및 음성합성을 위한 다국어 낭독 데이터로, 실제 녹음 음성과 텍스트가 포함된 1,000시간 분량의 대규모 공개 데이터이다[28]. 그림 18과 같이 9개 국어의 음성이 포함되어 있으며, MLAAD와 비슷한 개수로 맞추기 위해 9개의 언어에 대해 8,000개씩 총 72,000개를 랜덤 추출했다. 본 논문에서는 추가적인 테스트를 위해 KoAAD[29]를 사용해 학습 데이터에 포함되어 있지 않던 한국어에 대한 성능을 평가하였다.

MLAAD 및 M-AILABS의 약 140,000개의 데이터 중, 총 5,000개의 샘플을 추출하여 실험했다. 2,500개는 학습, 1,000개는 검증, 1,500개는 테스트로 사용했으며, 생성된 가짜 음성과 실제 음성의 비율은 거의 같도록 유지했다. 손실 함수는 이진 크로스 엔트로피, 배치 크기는 4, 학습률은 10^{-4}, 아담 옵티마이저를 사용하고 조기 멈춤을 적용하여 최대 10 에포크까지 학습을 진행하였다. 실행 환경은 구글 코랩(Google Colab)에서 T4 GPU를 사용해 진행했으며, 재현성을 위해 파이썬 랜덤 시드, numpy 시드, torch 시드, CUDA 시드를 매번 모두 42로 고정하고 학습을 진행하였다.

먼저 기준 모델(LCNN-LSTM)과 비교를 위한 기존 모델(Whisper-LCNN, SpecRNet, MesoNet)의 학습을 진행한다. 이후 제안된 각 기술을 개별로 적용한 모델 및 통합 적용 모델을 학습하여 결과를 분석한다. 테스트는 모델의 일반화 성능을 평가하기 위해 기본 테스트 데이터, 잡음 추가 테스트 데이터(데이터 증강), 한국어 테스트 데이터(KoAAD)에 대해 각각 진행한다.

2. 실험 결과 및 분석

모델의 성능은 정확도(Accuracy), F1-score, 동일 오류율(EER: Equal Error Rate), 최소 탐지 비용 함수(minDCF: minimum Detection Cost Function) 네 개의 지표를 통해 평가한다. 정확도는 전체 데이터 중 맞춘 데이터의 비율로 높을수록 좋다. F1-score는 정밀도(Precision)와 재현율(Recall)의 조화평균을 계산한다. 1에 가까울수록 성능이 좋다. 동일 오류율(EER)은 오인식률(False Acceptance Rate, FAR)과 오거부율(False Rejection Rate, FRR)이 같은 지점을 의미한다. 이는 딥보이스 탐지에서는 딥보이스를 실제 음성으로, 실제 음성을 딥보이스로 판단하는 정도를 나타내며, 낮을수록 성능이 좋다. 또한 딥보이스를 탐지하지 못하는 경우가 실제 음성을 딥보이스로 판단하는 경우보다 치명적이므로 최소 탐지 비용 함수를 통해 오인식률과 오거부율의 비중을 다르게 계산하여 실제 사용 시 적합

성을 판단하고자 했다. 테스트 데이터셋의 양성 비율은 0.5이며, 딥보이스를 실제 음성으로 분류했을 때의 비용을 2, 실제 음성을 딥보이스로 분류했을 경우의 비용을 1로 설정한 뒤 비용을 계산했다.

기본 테스트 데이터에 대한 성능은 표 1과 같다. 제안하는 기술을 적용한 종합 개선 모델이 정확도 99.67%로 가장 성능이 좋았다. 개선 방법 적용 이전에는 기본 모델인 LCNN의 성능이 SpecRNet이나 MesoNet보다 근소하게 좋았으며, Whisper-LCNN과는 탐지 성능이 거의 동일했다. 비교 모델들(Whisper-LCNN, SpecRNet, MesoNet)중에는 Whisper-LCNN의 성능이 가장 좋았다. 제안 기술 (A)만을 사용했을 경우, 약간의 성능 개선이 있었다. 이는 가정했던 것처럼 주파수 영역별로 가짜 오디오의 특징에 차이가 작게나마 존재한다는 것을 의미한다. 제안 기술 (B)만을 사용했을 경우에는 약간의 성능 하락이 존재했다. 제안 기술 (C)를 적용했을 경우, 성능이 소폭 개선되었다. 제안하는 3개의 기술을 모두 적용한 경우(Combine), 기존 기술들보다 우수한 성능을 나타냈다.

주변 잡음이 추가된 테스트 데이터에 대한 성능은 표 2와 같다. 이 표에서 알 수 있듯이, 주변 잡음이 추가된 데이터에 대해, 모든 모델들의 성능이 감소한 것을 확인할 수 있다. 아주 작은 크기의 노이즈나 환경음을 추가하기만 해도 성능 하락이 크다는 것을 확인할 수 있다. 이를 통해 오디오 도메인의 탐지가 잡음에 민감하며 일반화가 어려운 것을 확인할 수 있었다. 기존 데이터에 대한 테스트와 마찬가지로, 비교용 모델 중에서는 Whisper-LCNN이 91.58%로 가장 정확도가 높았다. 잡음이 추가된 데이터에서는 종합 개선 모델(A+B+C Combine)이 정확도, F1-score, EER, minDCF 모두 가장 좋은 성능을 보였다. 이는 다른 시스템과 비교했을 때 상대적으로 잡음과 주변 소음에도 강건하며, 모델의 일반화 능력이 개선된 것이라 해석할 수 있다.

잡음이 추가된 한국어 테스트 데이터(KoAAD)에 대한 성능은 표 4와 같다. 한국어에 대해서는 모든 모델의 성능이 상당히 하락한 것을 확인할 수 있었다. 이는 한국어가 학습 데이터에 존재하지 않았기 때문으로, 각 모델들은 학습하지 않은 새로운 공격에 대한 테스트를 수행하는 것과

표 2. 기본 데이터에 대한 실험 결과
Table 2. Table of normal data

Model(System)	Train dataset	Test dataset	Accuracy	F1	EER	minDCF
LCNN-LSTM(baseline)	MLAAD	MLAAD	99.22%	0.9924	0.0077	0.0139
Whisper-LCNN	MLAAD	MLAAD	99.35%	0.9937	0.0064	0.0114
SpecRNet	MLAAD	MLAAD	98.96%	0.9899	0.0105	0.0149
MesoNet	MLAAD	MLAAD	98.56%	0.9860	0.0140	0.0265
(A) HPF-LCNN-LSTM	MLAAD	MLAAD	99.41%	0.9943	0.0059	0.0090
(B) Mean-LCNN-LSTM	MLAAD	MLAAD	98.89%	0.9892	0.0109	0.0185
(C) Enhance-LCNN-LSTM	MLAAD	MLAAD	99.41%	0.9943	0.0057	0.0113
(A+B+C) Combine-LCNN-LSTM	MLAAD	MLAAD	99.67%	0.9968	0.0032	0.0051

표 3. 잡음 데이터에 대한 실험 결과
Table 3. Table of noise added data

Model(System)	Train dataset	Test dataset	Accuracy	F1	EER	minDCF
LCNN-LSTM(baseline)	MLAAD	Augmented MLAAD	84.14%	0.8199	0.1534	0.2951
Whisper-LCNN	MLAAD	Augmented MLAAD	91.58%	0.9118	0.0815	0.1589
SpecRNet	MLAAD	Augmented MLAAD	85.12%	0.8336	0.1440	0.2821
MesoNet	MLAAD	Augmented MLAAD	80.94%	0.7750	0.1842	0.3418
(A) HPF-LCNN-LSTM	MLAAD	Augmented MLAAD	87.40%	0.8630	0.1221	0.2264
(B) Mean-LCNN-LSTM	MLAAD	Augmented MLAAD	90.86%	0.9038	0.0885	0.1760
(C) Enhance-LCNN-LSTM	MLAAD	Augmented MLAAD	84.66%	0.8268	0.1483	0.2831
(A+B+C) Combine-LCNN-LSTM	MLAAD	Augmented MLAAD	92.49%	0.9223	0.0727	0.1432

표 4. 한국어 잡음 데이터에 대한 실험 결과
Table 4. Table of noise added Korean data

Model(System)	Train dataset	Test dataset	Accuracy	F1	EER	minDCF
LCNN-LSTM(baseline)	MLAAD	KoAAD	49.76%	0.5691	0.5085	0.6901
Whisper-LCNN	MLAAD	KoAAD	34.78%	0.4926	0.6636	0.8598
SpecRNet	MLAAD	KoAAD	42.60%	0.4389	0.5742	0.8588
MesoNet	MLAAD	KoAAD	42.99%	0.4975	0.5750	0.8042
(A) HPF-LCNN-LSTM	MLAAD	KoAAD	47.55%	0.5997	0.5368	0.6597
(B) Mean-LCNN-LSTM	MLAAD	KoAAD	53.28%	0.6246	0.4766	0.6036
(C) Enhance-LCNN-LSTM	MLAAD	KoAAD	47.03%	0.5238	0.5337	0.7540
(A+B+C) Combine-LCNN-LSTM	MLAAD	KoAAD	46.46%	0.5750	0.5455	0.6980

표 5. 한국어 잡음 데이터에 대한 실험 분석 (FNR 및 FPR)
Table 5. Table of Analysis on Korean Noise Data (FNR and FPR)

Model(System)	FNR (False Negative Rate)	FPR (False Positive Rate)
LCNN-LSTM(baseline)	36.33%	65.36%
Whisper-LCNN	39.25%	93.47%
SpecRNet	56.91%	57.93%
MesoNet	45.83%	69.17%
(A) HPF-LCNN-LSTM	24.58%	82.77%
(B) Mean-LCNN-LSTM	25.41%	69.90%
(C) Enhance-LCNN-LSTM	44.33%	63.19%
(A+B+C) Combine-LCNN-LSTM	30.5%	78.60%

같다. 비교용 세 모델(Whisper-LCNN, SpecRNet, MesoNet) 모두 50% 이하의 정확도를 보여주었다. 이는 랜덤하게 선택했을 경우의 기댓값 50%보다 낮은 수치이므로 사실상 딥보이스 검출이 불가능했음을 나타낸다. 특히 Whisper-LCNN의 성능 하락 폭이 두드러지는데, 이는 전처리로 사용되는 Whisper-tiny.en 모델이 영어에 대해서만 학습된 모델이기 때문에 그 외의 언어에서는 효과적이지 못한 것으로 생각된다. 반면 LCNN은 약 49%로 낮지만 소폭으로 우수한 성능을 나타냈다. 제안 기술 (B)를 적용했을 경우, 약간의 성능 증가가 나타났지만 그 외의 개선 방식은 큰 의미가 없었다. 지표가 50% 정도의 낮은 수치라면 사실상 무작위 선택과 같으며, 실제로 사용하기 어려운 시스템임을 뜻한다.

표 5는 성능이 가장 저조했던 한국어 잡음 데이터에 대해 분석한 내용이다. 모든 모델이 딥보이스 음성을 1로, 실제 음성을 0으로 라벨링 한 데이터에 대해 학습했기 때문에, 위음성률(FNR)이 딥보이스를 실제 음성으로 판단한 비율, 위양성률(FPR)이 실제 음성을 딥보이스로 판단한 비율을 나타낸다. SpecRNet을 제외한 대부분이 실제 음성을 딥보이스로 판단한 비율(FPR)이 훨씬 높았다. 물론 딥보이스를 실제 음성으로 판단한 비율(FNR)도 꽤나 높았지만, FPR이 더 두드러졌다. 이는 학습 시 포함되지 않았던 한국어 데이터에 대해 실제 음성임에도 딥보이스로 판단한 경우가 많았다는 것으로, 딥보이스의 특징이 나타나지 않더라도 학습한 적 없는 억양이나 발음에 대해 딥보이스로 판단하여 성능이 하락했음을 알 수 있다. 이를 통해 언어 및 억양의 차이가 딥보이스 탐지에 매우 큰 영향을 준다는 점을 다시금 확인할 수 있었다.

V. 결 론

딥보이스 탐지는 모델의 강건성과 일반화 성능이 중요하며, 새로운 공격에 대한 적응이 매우 중요하다. 본 연구에서는 기존에 잘 활용되지 않던 딥보이스 다국어 데이터셋을 활용하며, LCNN-LSTM에 세 가지 개선 기술들을 적용하

여, 다양한 기존 모델들과 비교해 딥보이스 탐지 성능을 평가하였다. 평가는 정확도, F1-score, 동일 오류율(EER), 최소 탐지 비용 함수(minDCF: minimum Detection Cost Function)를 지표로 사용하여 기본 다국어 테스트 데이터, 잡음이 추가된 다국어 테스트 데이터, 한국어 잡음 테스트 데이터에 대한 성능을 비교하였다. 기본 다국어 테스트 데이터에서는 개선 방식을 모두 적용한 모델이 모든 지표에서 가장 높은 성능을 보였으며, 비교용 모델 중에서는 Whisper-LCNN이 가장 우수하였다. 잡음이 추가된 데이터에서는 모든 모델의 성능이 감소하였으며, 이번에도 종합 개선 모델(A+B+C Combine)이 잡음 추가 데이터에 대해 가장 우수한 성능을 보였다. 이는 잡음과 주변 소음에 강건하며 모델의 일반화 능력이 향상되었음을 의미한다. 한국어 잡음 데이터에서는 모든 모델의 성능이 크게 하락하였으며, 특히 Whisper-LCNN의 성능 하락이 두드러졌다. 이는 사전학습 된 Whisper-tiny.en 모델이 영어에 특화되어 있기 때문으로 해석된다. 기본 백본 모델(LCNN-LSTM)은 약간 나은 성능을 보였으며, 제안 기술 (B)를 적용한 모델이 53%로 가장 높은 정확도를 보였지만, 여전히 낮은 수치로 실제 사용에는 한계가 있음을 확인하였다. 결론적으로, 본 연구는 여러 딥보이스 탐지 모델의 성능이 다양한 환경에서 다르게 나타남을 확인했으며, 다국어 및 잡음 환경에서의 성능 개선이 필요함을 시사한다. 처음 보는 언어에 대해 성능이 하락하는 것을 통해 데이터셋의 중요성, 특히 우리말에 대한 딥보이스 데이터셋의 필요성을 다시금 확인할 수 있었다.

참 고 문 헌 (References)

[1] J. Yi, C. Wang, J. Tao, X. Zhang, C. Y. Zhang, & Y. Zhao. “Audio Deepfake Detection: A Survey.”, arXiv:2308.14970 [cs.SD], 2023.
doi: https://doi.org/10.48550/arXiv.2308.14970

[2] Y. Wang, R. Skerry-Ryan, E. Battenberg, D. Stanton, J. Shor, Y. Xiao, Y. Jia, F. Ren, P. Nguyen, Z. Chen, X. Chen, N. Jaitly, R. Prabhavalkar, & R. Saurous. “Tacotron: Towards End-to-End Speech Synthesis.”, Interspeech, vol. 2017, no. 9, pp. 4006-4010, Mar. 2017.
doi: https://doi.org/10.21437/Interspeech.2017-1452

[3] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, & K. Kavukcuoglu. “WaveNet: A Generative Model for Raw Audio.”, Speech Synthesis Workshop, vol. 2016, no. 9, pp. 1-6, Sep. 2016.
doi: https://doi.org/10.48550/arXiv.1609.03499

[4] R. Prenger, R. Valle, & B. Catanzaro. “WaveGlow: A Flow-based Generative Network for Speech Synthesis.”, 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), vol. 2019, no. 10, pp. 3617-3621, Oct. 2018.
doi: https://doi.org/10.1109/ICASSP.2019.8683143

[5] J. Su, Z. Jin, & A. Finkelstein. “HiFi-GAN: High-Fidelity Denoising and Dereverberation Based on Speech Deep Features in Adversarial Networks.”, Proc. Interspeech 2020, vol. 2020, no. 9, pp. 4506-4510, Sep. 2020.
doi: https://doi.org/10.21437/Interspeech.2020-2143

[6] E. Casanova, J. Weber, C. D. Shulby, A. C. Junior, E. Gölge, & M. A. Ponti. “YourTTS: Towards Zero-Shot Multi-Speaker TTS and Zero-Shot Voice Conversion for Everyone.”, Proceedings of the 39th International Conference on Machine Learning, PMLR, 162, pp. 2709-2720, 2022.
doi: https://doi.org/10.48550/arXiv.2112.02418

[7] E. Casanova, K. Davis, E. Gölge, G. Göknar, I. Gulea, L. Hart, A. Aljafari, J. Meyer, R. Morais, S. Olayemi, & J. Weber. “XTTS: a Massively Multilingual Zero-Shot Text-to-Speech Model.”, Proceedings of INTERSPEECH 2024, 2024.
doi: https://doi.org/10.48550/arXiv.2406.04904

[8] H. Kameoka, T. Kaneko, K. Tanaka, & N. Hojo. “StarGAN-VC: Non-parallel Many-to-Many Voice Conversion Using Star Generative Adversarial Networks.”, 2018 IEEE Spoken Language Technology Workshop (SLT), vol. 2018, no. 6, pp. 266-273, June 2018.
doi: https://doi.org/10.1109/SLT.2018.8639535

[9] Y. Yang, Y. Kartynnik, Y. Li, J. Tang, X. Li, G. Sung, & M. Grundmann. “StreamVC: Real-Time Low-Latency Voice Conversion.”, Proceedings of ICASSP 2024, 2024.
doi: https://doi.org/10.48550/arXiv.2401.03078

[10] Y. Gao, R. Singh, & B. Raj. “Voice Impersonation Using Generative Adversarial Networks.”, 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), vol. 2018, no. 1, pp. 2506-2510, Feb. 2018.
doi: https://doi.org/10.1109/ICASSP.2018.8462018

[11] S. Pascual, A. Bonafonte, & J. Serrà. “SEGAN: Speech Enhancement Generative Adversarial Network.”, Interspeech, vol. 2017, no. 3, pp. 3642-3646, Mar. 2017.
doi: https://doi.org/10.21437/Interspeech.2017-1428

[12] H. Zhou, Z. Liu, X. Xu, P. Luo, & X. Wang. “Vision-Infused Deep Audio Inpainting.”, 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 283-292.
doi: https://doi.org/10.1109/ICCV.2019.00037

[13] Z. Wu, R. K. Das, J. Yang, & H. Li. “Light Convolutional Neural Network with Feature Genuinization for Detection of Synthetic Speech Attacks.”, Interspeech, 2020, pp. 2817-2821.
doi: https://doi.org/10.21437/interspeech.2020-1810

[14] K. He, X. Zhang, S. Ren, J. Sun, “Deep Residual Learning for Image Recognition”, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2016, no. 90, pp. 770-778, December 2015.

doi: https://doi.org/10.1109/cvpr.2016.90
[15] P. Kawa, M. Plata, & P. Syga. “SpecRNet: Towards Faster and More Accessible Audio DeepFake Detection.”, 2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), 2022, pp. 792-799.
doi: https://doi.org/10.1109/TrustCom56396.2022.00111
[16] D. Afchar, V. Nozick, J. Yamagishi, & I. Echizen. “MesoNet: a Compact Facial Video Forgery Detection Network.”, 2018 IEEE International Workshop on Information Forensics and Security (WIFS), 2018, pp. 1-7.
doi: https://doi.org/10.1109/WIFS.2018.8630761
[17] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, “Robust Speech Recognition via Large-Scale Weak Supervision,” arXiv preprint arXiv:2212.04356, Dec. 2022.
doi: https://doi.org/10.48550/arXiv.2212.04356
[18] P. Kawa, M. Plata, M. Czuba, P. Szymański, & P. Syga. “Improved DeepFake Detection Using Whisper Features.”, ArXiv, vol. abs/2306.01428, 2023.
doi: https://doi.org/10.48550/arXiv.2306.01428
[19] ASVspoof2019, https://www.asvspoof.org/index2019.html
[20] G. Lavrentyeva, S. Novoselov, A. Tseren, M. Volkova, A. Gorlanov, & A. Kozlov. “STC Antispoofing Systems for the ASVspoof2019 Challenge.”, arXiv, 2019. URL: https://arxiv.org/abs/1904.05576
doi:10.48550/arXiv.1904.05576
[21] ASVspoof2021, https://www.asvspoof.org/index2021.html
[22] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, “Going deeper with convolutions”, 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2015, pp. 1-9, September 2014.
doi: https://doi.org/10.1109/CVPR.2015.7298594
[23] Y. Wang, M. Zhu, N. W. Campbell, & J. Wang. “High-Resolution Image Synthesis and Semantic Editing with Generative Adversarial Networks.” 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1704-1713, Jun. 2018.
doi: https://doi.org/10.1109/CVPR.2018.00186
[24] AI-HUB Virtual Space Ambient Sound Matching Data, https://aihub.or.kr/aihubdata/data/view.do?currMenu=&topMenu=&aihubDataSe=data&dataSetSn=71354
[25] piotrkawa/deepfake-whisper-features, https://github.com/piotrkawa/deepfake-whisper-features/tree/main/src/models
[26] N. M. Muller, P. Kawa, W. H. Choong, E. Casanova, E. Golge, T. Muller, P. Syga, P. Sperl, K. Bottinger, “MLAAD: The Multi-Language Audio Anti-Spoofing Dataset”, arXiv.org, vol. abs/2401.09512, pp. -, January 2024.
doi: https://doi.org/10.48550/arXiv.2401.09512
[27] MLAAD(Multi-Language Audio Anti-spoofing Dataset), https://owncloud.fraunhofer.de/index.php/s/tL2Y1FKrWiX4ZtP?path=%2Fv3
[28] M-AILABS speech dataset, https://www.caito.de/2019/01/03/the-m-ailabs-speech-dataset/
[29] KoAAD, https://github.com/ldh-Hoon/KoAAD

저 자 소 개

임 동 훈

- 2019년 ~ 현재 : 세종대학교 전자정보통신공학과 학사과정
- ORCID : https://orcid.org/0009-0005-8900-7202
- 주관심분야 : 음성처리, 음성합성, 자연어처리

한 종 기

- 1992년 : KAIST 전기및전자공학과 공학사
- 1994년 : KAIST 전기및전자공학과 공학석사
- 1999년 : KAIST 전기및전자공학과 공학박사
- 1999년 3월 ~ 2001년 8월 : 삼성전자 DM연구소 책임연구원
- 2001년 9월 ~ 현재 : 세종대학교 전자정보통신공학과 교수
- 2008년 9월 ~ 2009년 8월 : University California San Diego (UCSD) Visiting Scholar
- ORCID : https://orcid.org/0000-0002-5036-7199
- 주관심분야 : 비디오 코덱, 영상 신호처리, 정보 압축, 방송 시스템