데이터과학 및 분석

Fake Audio Detection

DATA SCIENCE

컴퓨터공학부 22학번

이성준

오승진, 이호성

20223179, 20223182

데이터과학 및 분석

Fake Audio Detection

TABLE OF CONTENTS

개요

<!-- 프로젝트명 -->	• Fake Audio Detection
<!-- 배경 -->	• AI 기술이 급속도로 발전하면서 페이크 미디어 급증
<!-- 목표-->	• 다양한 환경에서 모방 혹은 녹음된 음성 탐지
<!--개발 환경 -->	• Colab A100, Pycharm 4060Ti, Anaconda, Python
<!--개발 환경 -->	• Tensorflow,

데이터 및 특징

데이터 구성

for-2sec (for-2seconds)
```
├── training
│   ├── real
│   └── fake
├── validation
│   ├── real
│   └── fake
└── testing
    ├── real
    └── fake
```

for-norm
```
├── training
│   ├── real
│   └── fake
├── validation
│   ├── real
│   └── fake
└── testing
    ├── real
    └── fake
```

for-original
```
├── training
│   ├── real
│   └── fake
├── validation
│   ├── real
│   └── fake
└── testing
    ├── real
    └── fake
```

for-rerec
```
├── training
│   ├── real
│   └── fake
├── validation
│   ├── real
│   └── fake
└── testing
    ├── real
    └── fake
```

데이터 출처

Kaggle :
The Fake-or-Real (FoR) Dataset (deepfake audio)
Mohammed Abdeldayem · Abdalla Mohamed
라이센스 : GNU Lesser General Public License
오픈소스 데이터셋

데이터 구성

for-2sec (for-2seconds): 발화가 2초로 잘린 버전
for-norm: 성별 및 클래스 균형이 맞춰진 표준화된 데이터셋
for-original: 원본 데이터, 표준화 없음
for-rerec: 재녹음된 데이터로 음성 채널 공격 시나리오 재현

사용 데이터

for-norm, for-2sec, for-rerecorded를 4.5초로 패딩
초가 짧은 경우 반복하고 넘을 경우 자름

전처리

Preprocess

TABLE OF CONTENTS

초기 계획 - > 전처리 방법

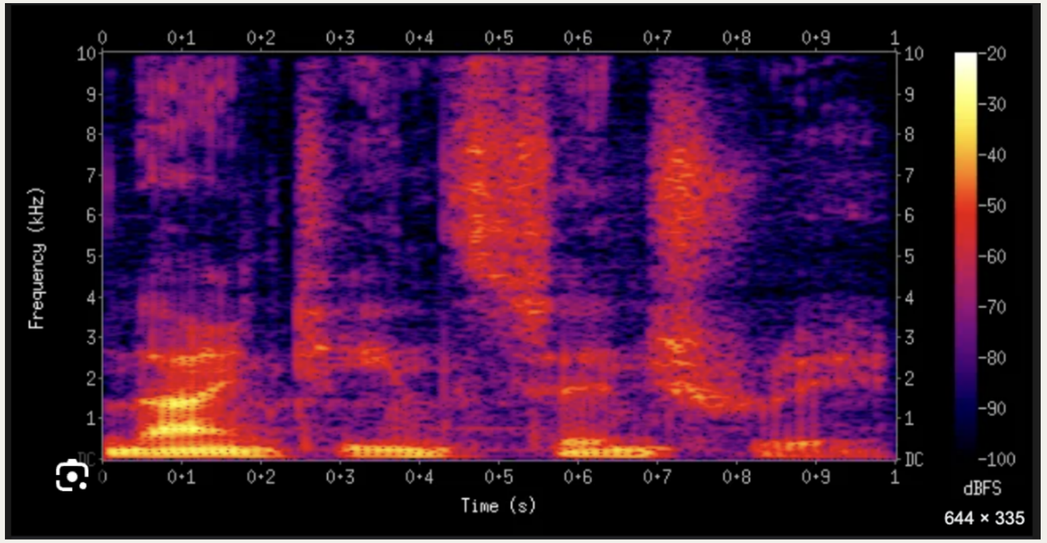
모델 선정

초기 계획

✳ 이미지 데이터

- 스펙트로그램 :

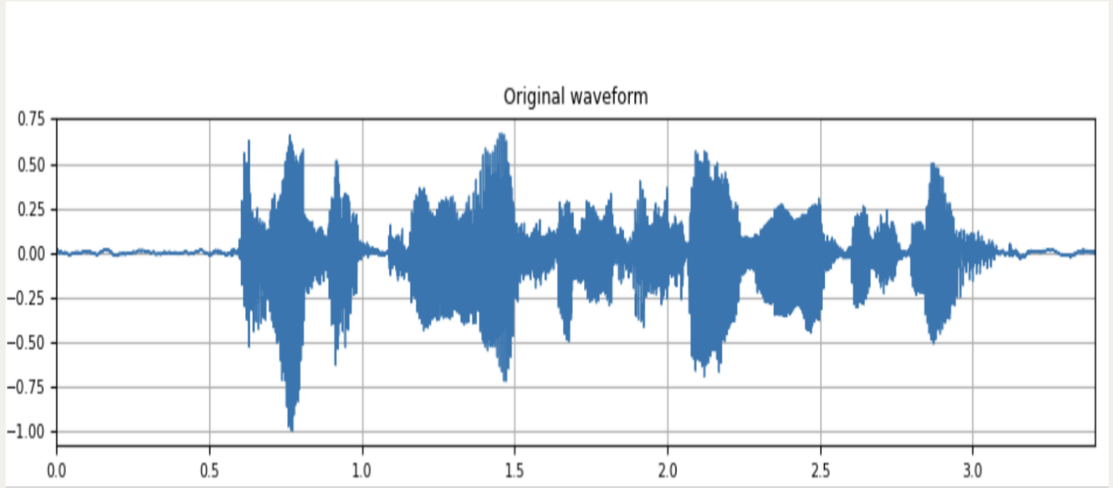
  MelSpectrogram, Spectrogram을 추출 후 학습



✳ 시간 축에 따라 배열된 특징 벡터

- 특징 추출 :

  MFCC, LFCC를 추출한 후 CSV 데이터로 만든 후 데이터프레임 변환하여 학습



최종 전처리 방법

✳ 시간 축에 따라 배열된 특징 벡터

왜 이 방법을 택했는가? :

1. 자원의 한계
    1. 코랩의 지속적인 비용을 지불해야 함
    2. 로컬 리소스의 한계

2. 월등한 학습 속도
    1. 이미지 데이터의 비해 학습 속도가 빠름
    2. 성능 측면에서도 이미지 데이터보다 성능이 잘 나옴

3. 시퀀스 형성
    1. 시퀀스 형성을 통해 시계열 특성을 잘 학습할 수있음

E	F	G	H	I	J	K	L	M	N	
0	1	2	3	4	5	6	7	8	9	
-425.16995	68.38909	20.780706	-1.8245784	-33.90544	-9.71883	-15.69289	-20.923552	-8.18415	-9.442834	-2
-332.16803	116.83399	9.640101	75.8551	19.165907	-1.948786	5.756593	-13.329574	-10.208344	-10.173299	-1
-364.27994	33.17641	14.797523	19.127687	-40.804897	-20.301262	-31.19606	-25.65114	-10.828776	-15.588645	-
-367.30093	39.572197	12.220928	12.33511	-16.90208	-20.383379	-29.227953	-20.15562	-6.751601	-16.504707	
-324.7535	97.85393	-26.91479	40.901424	17.526657	-17.666306	7.789697	-1.5166541	-7.5140753	22.225172	-
-329.61755	93.98397	1.9938443	2.6165087	-24.118145	-6.108116	-24.289295	-4.619783	-22.75677	-15.876464	-
-432.99	70.687965	8.206307	19.467146	-21.51723	-20.075527	-20.695461	-18.256655	-5.6461816	-16.566229	
-328.4271	107.61788	-5.760798	59.237633	23.755007	10.476191	2.4482512	-4.594166	8.400594	-1.0191861	-1
-305.11362	98.91478	-8.188933	46.25279	-11.668834	-4.231655	0.9829397	-22.73539	-16.610632	-4.9425006	-3
-408.53824	71.014626	-12.259116	-3.5858226	-24.841042	-32.359528	-18.46207	-4.9105654	-17.157272	-3.2278986	-1
-341.39145	90.30765	1.1236447	27.012115	-33.352524	-19.263992	-28.423822	-18.856441	-11.445041	-11.65956	-3
-342.32236	99.98642	2.820423	59.847332	-5.7473164	-0.5831714	14.47164	-5.0382066	1.1943817	-2.8765805	-
-340.87048	91.96725	16.323874	32.370056	-13.311891	-2.9214313	-26.752312	0.16329522	-29.70634	9.40188	-
-404.16202	103.201706	-2.034728	8.892078	-25.563255	-4.5887456	-22.489977	-19.796799	0.25924957	-16.863466	-
-311.25894	105.62752	-8.613918	73.03429	4.858539	12.444948	-22.13941	-2.1539316	-26.22989	3.0345564	-1
-318.99963	145.76231	-19.016356	45.643177	-0.38151315	-14.099002	14.524715	-10.586656	-3.6840372	1.1198004	-
-369.35522	125.99098	-81.260666	23.289686	-27.694704	-41.93673	-19.550404	-10.255036	-14.180073	1.862688	-6
-453.39688	75.3867	13.880962	33.433308	18.067904	16.13345	10.275534	18.23974	10.258079	9.918136	1
-436.19464	58.28062	-6.397666	47.314655	-15.224179	36.13258	-12.346223	27.601921	-3.769308	15.139186	-4
-341.10968	29.60198	-53.592552	56.567783	-26.206589	34.75807	-17.409208	21.590387	-17.122683	13.986254	-4
-380.7751	128.90817	3.7415113	22.983614	12.526986	20.133896	10.741498	1.4905846	2.787147	-6.1253133	1
-370.1307	72.92874	13.523958	14.821046	-26.9685	-25.761911	-20.281982	-7.2179856	-8.780474	-18.33518	-
-429.5407	64.465034	8.758934	1.4844104	-20.275854	-15.11619	-21.932144	-15.117428	-5.7627773	-16.735434	-6

이미지 출처 - 나무위키, Pytorch

최종 전처리 방법

✳ 이미지 데이터

이미지 데이터를 제외 한 이유 :

1. 학습 속도

1. 복잡한 모델 구조를 가지면, 파라미터 수 가 늘어나면서 에포크 당 처리 속도가 급격히 증가함

2. 테스트 데이터 성능

1. MFCC와 LFCC를 학습한 것에 비해 테스트 데이터의 성능이 30% 이상 차이가 남
2. 학습과 검증세트에서 과적합의 징후를 전혀 보이지 않지만 테스트 성능에서 과적합을 보임

→

이를 해결하기 위해 뭘 했는가?

1. 전이학습 및 단계 학습

1. Mixture of Experts 방법론을 통한 전이학습
   - 실제 테스트 성능에서 효과를 못봄
   - 적은 데이터 셋으로도 에포크 당 5~10분 정도 소요됨
2. 단계 학습
   - 데이터 셋 별로 나눠서 학습할 시 Catastrophic Forgetting 발생 우려
3. 샘플링 학습
   - 데이터의 수가 적을 때의 성능과 많을 때의 성능이 같다는 보장을 할 수가 없음

모델 선정

Choose Models

TABLE OF CONTENTS

GRU, LSTM, TRANSFORMER

최종 모델 구조

✳ LSTM

구조 특징: Conv1D 기반 초기 특징 추출 후 Multi-Head Attention과 Feedforward로 구성된 TransformerBlock을 반복.
장점: 장기적 의존성을 효과적으로 학습하며 복잡한 데이터 구조에 적합.

✳ GRU

구조 특징: Bidirectional LSTM으로 시퀀스를 처리하고 정규화와 드롭아웃을 추가.
장점: 시간적 종속성(Time Dependency)을 잘 학습하며, 데이터가 적은 경우에도 안정적인 성능.

✳ Transformer

구조 특징: LSTM과 유사하지만 더 단순한 GRU로 구성되며, 양방향 처리와 정규화를 적용.
장점: LSTM보다 훈련이 빠르고 계산 효율이 높아 경량 모델 구현에 적합.

✳ 앙상블

구조 특징: 각 모델의 성능에 따라 최종 예측에 반영하는 앙상블 기법
장점: 더 성능이 좋은 모델에 더 높은 가중치를 줘서 단순 다수결보다 정확한 예측이 가능

학습 및 검증과 테스트
Train, Validation, Test

TABLE OF CONTENTS

학습 횟수, 평가 지표, 테스트 방법

학습 횟수, 평가 지표, 테스트 방법

01

학습 횟수

모델 별로 에포크(epoch)를 최대 100회로 지정하고 Validation Loss가 5회 이상 개선되지 않으면 학습을 종료시킴

02

평가지표

혼동행렬, F1 Score, Binary Crossentropy 손실 함수를 통해 과적합 여부와 성능을 평가함

03

테스트 방법

데이터 셋이 학습, 검증, 테스트 셋으로 분할이 되어 있으므로 테스트 세트에 대한 성능을 분석

결과 보고

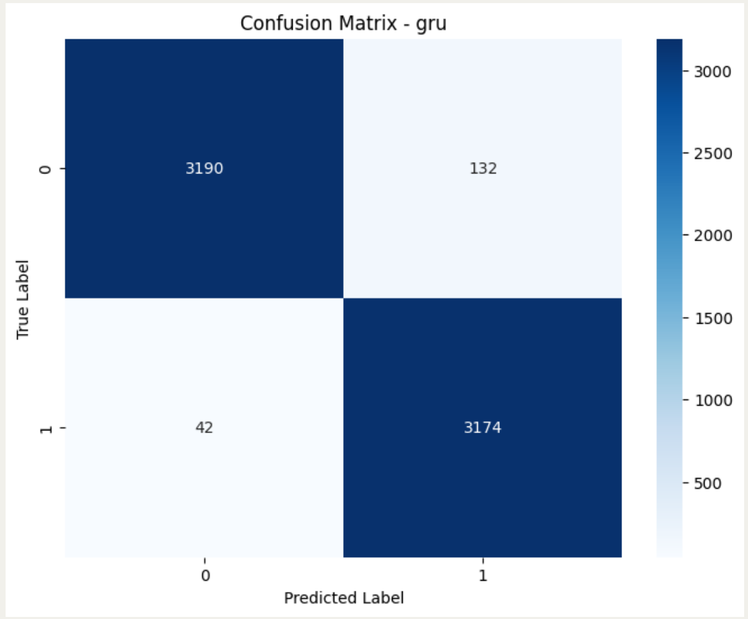
Result Report

TABLE OF CONTENTS

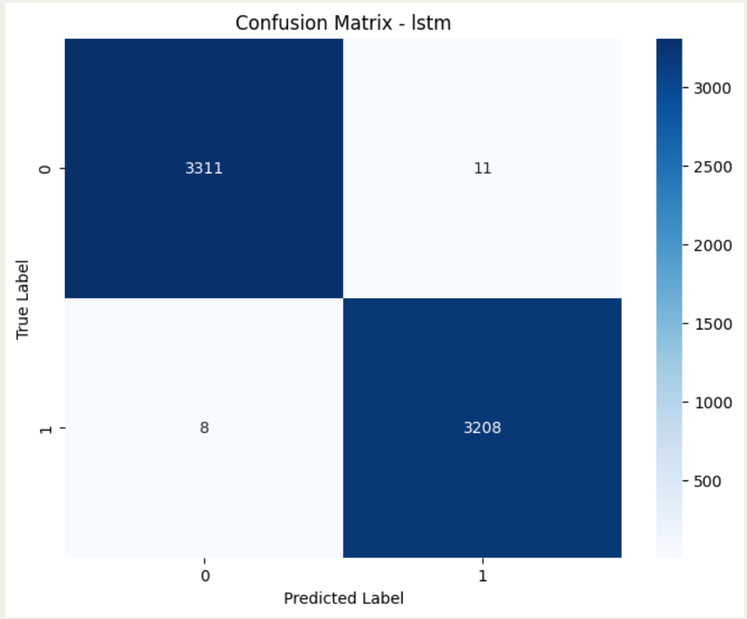
앙상블 혼동행렬

모델별 결과 보고

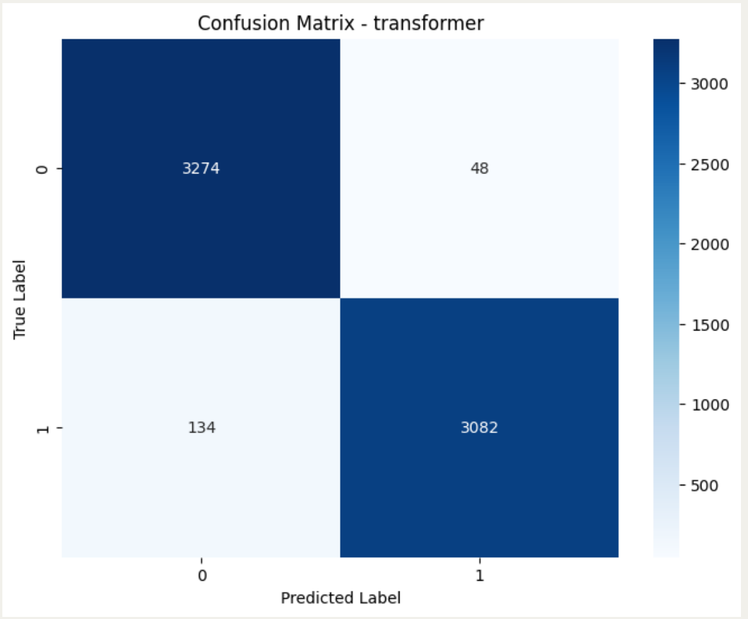
GRU



LSTM



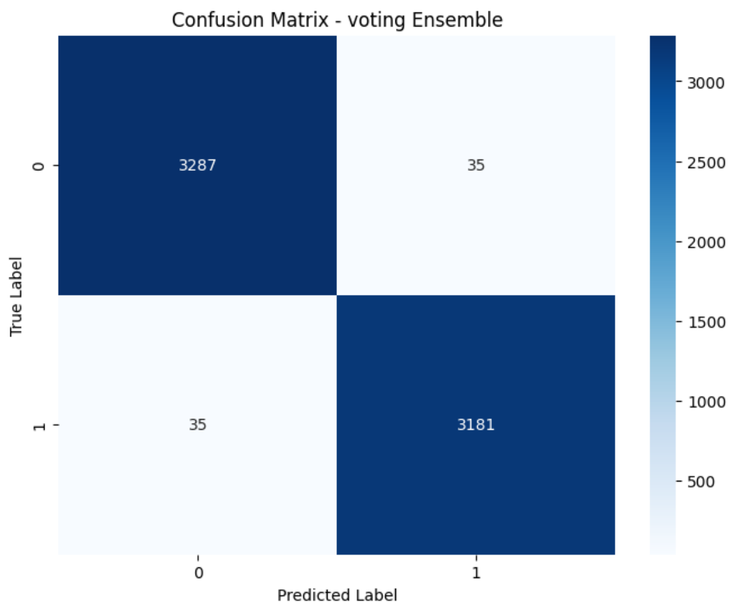
Transformer



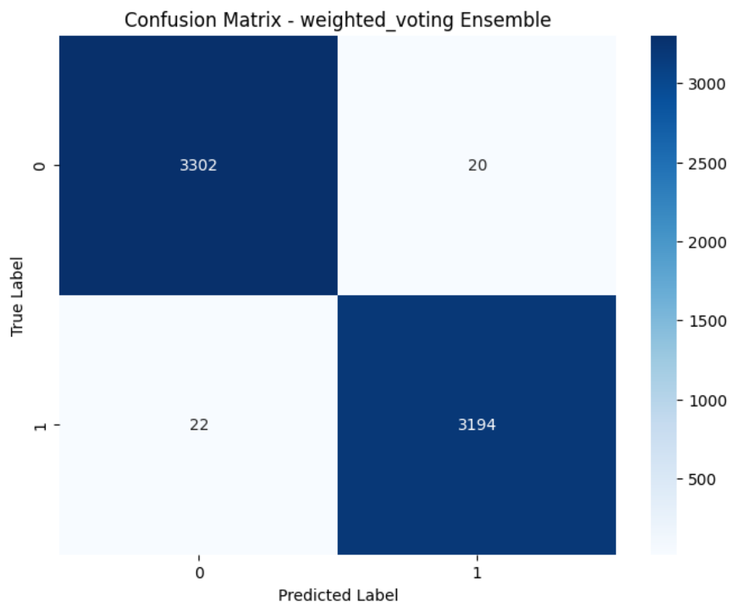
LSTM이 우수한 성능을 보이지만 조금의 향상을 위해 앙상블 시도

앙상블 결과 보고

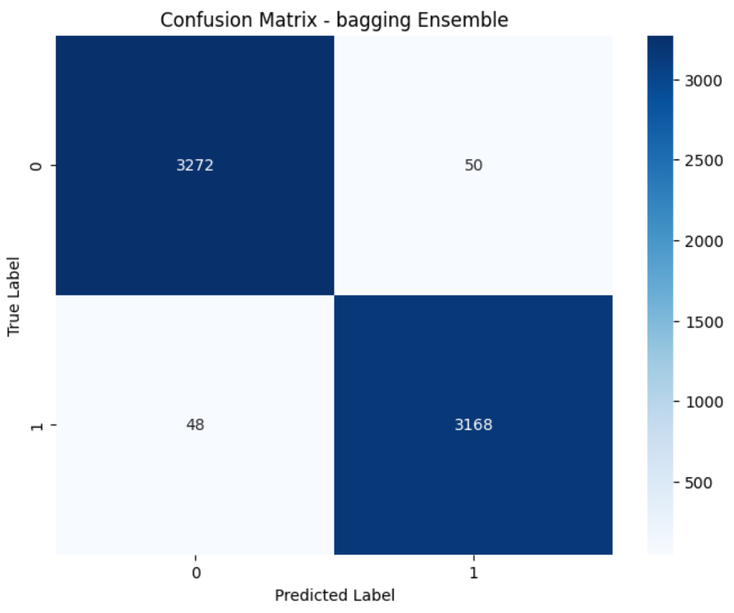
Voting



WeightedVoting



Bagging



Lstm = 0.5 / Gru = 0.3 / Transformer = 0.2

출처

참고 문헌

김, 소운, & 이, 성택. (연도). 딥보이스를 악용한 보이스 피싱 피해방지 서비스 개발. 한국정보통신학회논문지, 47(10), 1456-1463. doi:10.1234/jkics.2020.47.10.1456

박, 대서, 방, 준일, 김, 화종, & 고, 영준. (2020). CNN을 이용한 음성 데이터 성별 및 연령 분류 기술 연구. 한국정보통신학회논문지, 45(3), 210-220.

https://github.com/sksmta/audio-deepfake-detection

Mixture of Experts Fusion for Fake Audio Detection Using Frozen wav2vec 2.0
Zhiyong Wang, Ruibo Fu, Zhengqi Wen, Jianhua Tao, Xiaopeng Wang, Yuankun Xie, Xin Qi, Shuchen Shi, Yi Lu, Yukun Liu, Chenxing Li, Xuefei Liu, Guanjun Li

Q&A