

[서식 5] 캡스톤디자인 최종보고서

캡스톤디자인 최종보고서								
과제명		딥보이스를 이용한 음성 사기 예방을 위한 애플리케이션 개발						
팀명		Clean Code			수행기간		2024학년도 1학기	
교과담당교수	소속	소프트웨어학부			성명		이현준	
팀지도교수	소속	소프트웨어학부			성명		유홍석	
필드 마스터	소속	아이커넥트			전화		010-9728-1993	
	성명	장재석			E-mail		jaeseok.jang@gmail.com	
참여기업		기업명		아이커넥트		대표자		장재석
구분		성명	학과	학번	학년	휴대전화	E-mail	
참여 학생	팀장	이현준	소프트웨어학부	201805013	4	010-4940-9501	bmj12@naver.com	
	팀원 <small>*필요시확장</small>	서창희	소프트웨어학부	201905031	4	010-6644-8907	wldnis3608@naver.com	
		순현상	소프트웨어학부	201905013	4	010-4819-7811	tnsgustkd@naver.com	
		조성훈	소프트웨어학부	201905057	4	010-4132-0445	mc213213@naver.com	
		강지윤	소프트웨어학부	202105011	4	010-5165-5658	wldbs5165@naver.com	
	과제유형		과제분류	<input type="checkbox"/> 융합형 <input type="checkbox"/> 기업연계형 <input type="checkbox"/> 지역사회연계형 <input type="checkbox"/> 창업연계형 <input type="checkbox"/> 글로벌 <input type="checkbox"/> 학생창작형				
과제유형			<input type="checkbox"/> 기획 및 아이디어 <input type="checkbox"/> 작품(시작제품)개발 <input type="checkbox"/> 학술연구					
성과형태			<input type="checkbox"/> 지적(지식)재산권 출원 <input type="checkbox"/> 경진대회 수상/응모 <input type="checkbox"/> 아이디어 및 기술 양도 <input type="checkbox"/> 아이디어 및 기술 판매(육선) <input type="checkbox"/> 재능기부 <input type="checkbox"/> 취업 및 창업					
본 보고서를 경운대학교 SW중심대학사업단 규정에 의해 과제를 성실하게 수행하여 보고서를 제출합니다.								
서 식 3. 최종보고서 1부. 서 식 4. FM지도일지 1부. 서 식 4-1. FM증빙사진 1부.								
2024. . .								
교과목담당교수 : 정용환 (서명 또는 인) 팀지도교수 : 유홍석 (서명 또는 인) 대표학생 : 이현준 (서명 또는 인)								
경운대학교 소프트웨어중심대학사업단장 귀하								

- 참여 학생 작성 부문 -

1. 과제 개요

가. 과제 선정의 배경



딥보이스의 접근성과 기술력이 향상함에 따라 보이스 피싱에 악용되는 사례가 늘어나고 있다. 그림 1과 같이 지인의 목소리를 학습하여 지인과 같은 목소리로 돈을 요구하여 돈을 보내 피해자가 생기는 사례가 점점 늘어나고 있다. 그리고 그림 2와 같이 60대 이상이 보이스 피싱에 당하는 사례가 가장 많으며 한번 보이스 피싱 사기를 당할 때의 평균 금액이 증가하고 있다. 개인뿐만 아니라 기업도 딥보이스를 이용한 보이스 피싱에 피해를 입고 있다.



그림 3,4는 구글 스토어에 각각 딥보이스 판별, 보이스 피싱을 검색했을 때 나오는 애플리케이션들이다. 딥보이스 판별을 검색했을 경우에는 판별이 아닌 딥보이스 음성을 만드는 애플리케이션만 출력이 되고 보이스 피싱을 검색을 하면 딥보이스를 이용한 보이스 피싱 판별 기능을 제공하는 애플리케이션을 접하기 힘들다. 이와 같이 사용자들이 딥보이스 이용한 보이스 피싱 판별 기능을 제공하는 애플리케이션을 접하기 힘든 지금 우리는 딥보이스와 일반 음성의 음성 특징을 추출하여 모델을 학습해 사용자가 보이스 피싱인지 아닌지 판별 가능한 애플리케이션 개발을 목표로 선정하였다.

나. 과제 개발 혹은 제작에 따른 기대효과

하나의 음성 특징 추출 알고리즘 사용이 아닌 두 개의 음성 특징 추출 알고리즘을 사용하여 생성된 딥러닝 모델을 앙상블 기법 중에 소프트 보팅을 이용하여 하나의 음성 특징 추출 알고리즘을 사용했을 때 보다 정확한 판별이 가능할 것이라 기대되고 사용자에게 성별 정보와 같은 카테고리를 애플리케이션 선택 가능하게 하여 카테고리에 해당하는 딥러닝 모델을 적용하여 판별을 해 정확도를 높일 수 있을 것이라 기대된다. 애플리케이션 개발함으로써 사용자가 휴대폰에 다운받기만 하면 보다 쉽게 서비스를 제공할 수 있을 것이라 생각한다. 위와 같은 서비스 제공하는 애플리케이션을 개발함으로써 딥보이스를 이용한 보이스 피싱이 늘어나고 있는 지금 사용자가 사기를 당하기 전에 딥보이스를 이용한 보이스 피싱인지 아닌지 판별하여 사전 예방이 가능하게 하여 피해 사례나 금액이 감소 할 것이라 기대된다.

2. 활용 / 관련 이론

구 분	관련 이론 및 지식
전공 분야	librosa에서 제공하는 MFCC, Mel-Spectrogram을 이용한 음성 특징 추출, Keras에서 제공하는 VGG-19, BiLSTM을 이용한 모델 생성 Flask을 이용한 서버 구축, Flutter을 이용한 애플리케이션 개발 Anaconda를 이용한 가상 환경 구축

3. 조원별 역할분담

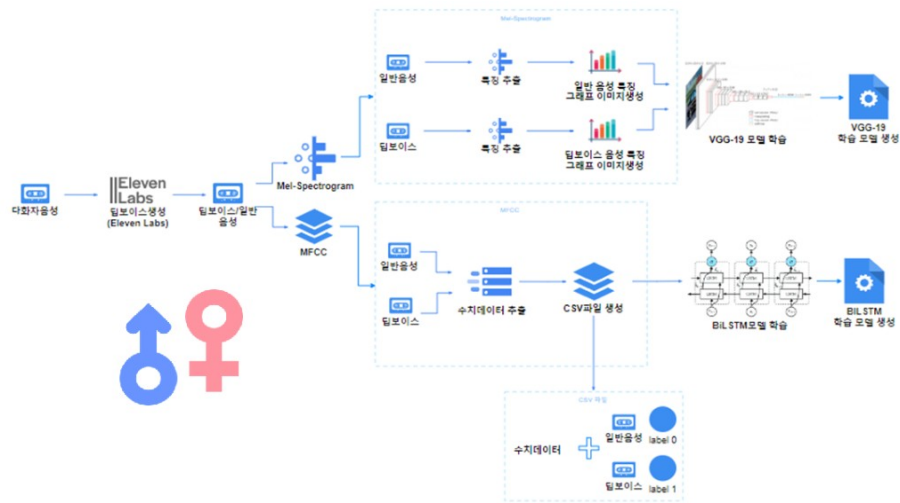
No.	성명	담당	수행 역할
1	이현준	팀장 및 총괄, 음성 특징 추출, 학습/검증 데이터 확보	팀 총괄, 전체적인 프로그램 기획 및 개발, MFCC와 Mel-Spectrogram을 통한 음성 특징 추출, 학습/검증 음성 데이터 생성 및 확보
2	서창희	VGG-19 모델 생성, 학습/검증 데이터 확보	음성 그래프 이미지를 사용해 VGG-19 학습 모델 생성, 소프트 보팅 적용 학습/검증 음성 데이터 생성 및 확보
3	순현상	애플리케이션 기능, UI 개발	애플리케이션 기능 및 UI 개발
4	조성훈	BiLSTM 모델 생성, 학습/검증 데이터 확보	수치 데이터를 이용한 BiLSTM 학습 모델 생성, 소프트 보팅 적용 학습/검증 음성 데이터 생성 및 확보
5	강지윤	애플리케이션 기능, UI 개발	애플리케이션 기능 및 UI 개발

4. 과제의 수행 과정

추진 내용	책임자	월				월				월				월	
		1주	2주	3주	4주	1주	2주	3주	4주	1주	2주	3주	4주	1주	2주
자료조사	이현준	■	■												
구성도 설계	이현준		■	■											
딥보이스 생성	조성훈			■	■	■	■	■	■						
음성 특징 추출	이현준						■	■	■	■					
모델 학습	서창희				■	■	■	■	■	■	■				
양상블 기법	조성훈											■	■		
서버, 애플리케이션 개발	강지윤, 순현상							■	■	■	■	■	■		
테스트	서창희											■	■	■	■

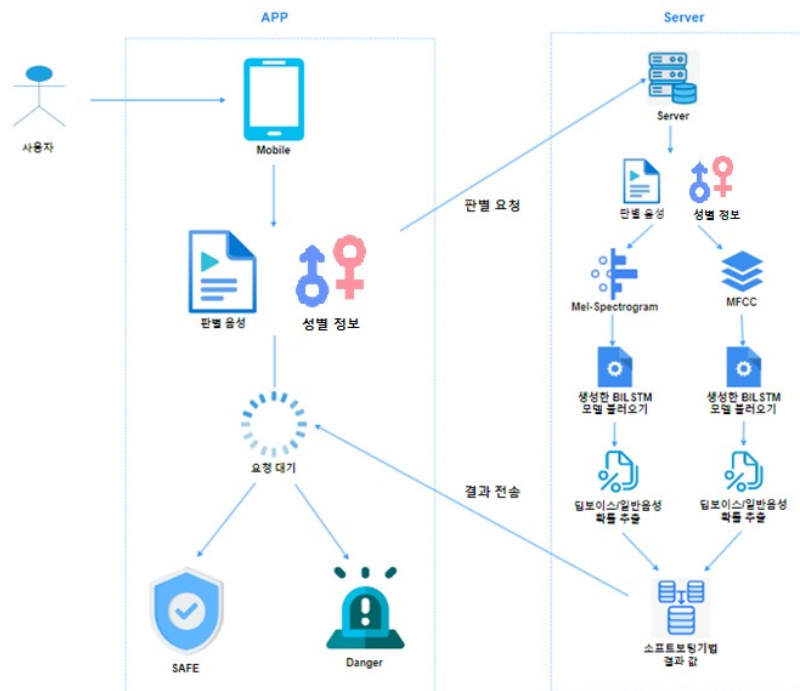
5. 세부 구성내용

1) 전체 흐름도 1.1 딥러닝 모델 생성



AI-hub에서 제공하는 음성 데이터를 이용해 일반 음성과 딥보이스를 생성할 음성을 확보하였고 ElevenLabs를 통해 딥보이스를 생성하였다. 그래서 총 남/여 각각 일반 음성 1000개 딥보이스 음성 1000개를 이용해서 모델을 생성하였다. 음성 특징 추출에는 음성 특징 추출에는 수치 데이터 형식의 특징 추출인 MFCC와 그래프 이미지 데이터 형식의 특징 추출인 Mel-Spectrogram을 이용하여 특징 추출을 하였다. 딥러닝 모델 생성에서는 이미지 형식의 학습에 강점이 있는 VGG-19모델과 시계열 데이터 학습에 강점이 있는 BiLSTM모델을 이용하여 남/여 분리하여 모델을 각각 생성하였다.

1.2 서버, 애플리케이션 동작



사용자는 애플리케이션을 통해 판별 음성과 성별 정보를 서버에 전송을 하고 받은 판별 음성과 성별 정

보를 바탕으로 성별 정보에 맞는 모델을 불러와 음성을 판별 한다. 그 후 소프트 보팅을 통해 최종 결과를 추출하고 애플리케이션으로 전송 후 결과에 맞는 화면을 인터페이스에 출력한다.

2) 프로그래밍 코드

2.1 MFCC 특징 추출

MFCC 특징 추출 함수

```
def extract_features(file_path):
    y, sr = librosa.load(file_path, sr=16000)
    mfcc = librosa.feature.mfcc(y=y, sr=sr, n_mfcc=100, n_fft=400, hop_length=160)
    return mfcc.T
```

그림 7 MFCC 특징 추출 함수

음성 특징 추출을 위해 1초에 16000개의 샘플링 주파수를 취한다. MFCC 특징 추출 설정 값은 0.01초 동안 400개의 샘플링 주파수 중에 100개의 음성 특징 값을 추출한다. 추출한 수치데이터는 csv파일로 저장되어 진다.

2.1 Mel-Spectrogram 특징 추출

Mel spectrogram을 추출하는 함수

```
def extract_mel_spectrogram(file_path, save_path):
    y, sr = librosa.load(file_path, sr=16000) # 음성 파일 로드
    mel_spectrogram = librosa.feature.melspectrogram(y=y, sr=sr, n_fft=400, hop_length=160, n_mels=128)
    log_mel_spectrogram = librosa.power_to_db(mel_spectrogram, ref=np.max) # 로그 스케일 변환

    # 그래프 그리기 및 저장
    plt.figure(figsize=(10, 4))
    librosa.display.specshow(log_mel_spectrogram, sr=sr, hop_length=160, cmap='gray', x_axis='time', y_axis='mel')
    plt.colorbar(format="%+2.0f dB")
    plt.savefig(save_path)
    plt.close()
```

그림 8 Mel - Spectrogram 추출 함수

MFCC와 똑같은 특징 설정 값을 준다. 그 후 특징 한 특성 값들을 power_to_db를 통해 데시벨로 값으로 변환해준다. 그러면 더 넓은 범위의 음량을 효과적으로 다룰 수 있다. 그 후 데시벨 값을 바탕으로 그래프를 그려서 음성 특징을 갖는 그래프 이미지를 생성한다.

2.2 BiLSTM 딥러닝 모델 생성

```
def create_bilstm_model(input_shape):
    model = Sequential()

    # 첫 번째 Bidirectional LSTM 층
    model.add(Bidirectional(LSTM(units=512, return_sequences=True), input_shape=input_shape))
    model.add(Dropout(0.5))

    # 출력층
    model.add(Dense(1, activation='sigmoid'))

    return model
```

Bidirectional는 양방향 LSTM 층을 의미한다. 유닛 수는 512개로 설정되어 있고 return_sequences 를 True로 설정해 출력으로 시퀀스를 반환하여 양방향으로 처리하여 순방향 및 역방향둘다 처리 함으로써 Bilstm모델을 생성한다. 그리고 과적합 방지를 위하여 Dropout은 0.5로 설정해 줬다. 그리고 출력층은 Dense로 선언해 주었다.

2.3 VGG-19 딥러닝 모델 생성

```
# 데이터 제네레이터 생성
batch_size = 32
img_height = 128
img_width = 275

datagen = ImageDataGenerator(rescale=1./255, validation_split=0.3)

train_generator = datagen.flow_from_directory(
    directory='D:/mels/train',
    target_size=(img_height, img_width),
    batch_size=batch_size,
    class_mode='categorical',
    subset='training'
)

val_generator = datagen.flow_from_directory(
    directory='D:/mels/train',
    target_size=(img_height, img_width),
    batch_size=batch_size,
    class_mode='categorical',
    subset='validation'
)
```

VGG-19모델 생성 코드 동작 시 할당되는 GPU와 메모리량이 많아 코드 실행이 되지 않는다. 그래서 Keras에서 제공하는 제너레이터를 사용해 학습에 필요할 때 마다 데이터를 불러오는 식으로 코드를 수정해 줬다. 경로안에 일반 음성과 딥보이스 음성 폴더가 있으면 자동으로 라벨링하는 기능을 제공한다.

```
base_model = VGG19(weights='imagenet', include_top=False, input_shape=(img_height, img_width, 3))
for layer in base_model.layers:
    layer.trainable = False

model = Sequential([
    base_model,
    Flatten(),
    Dense(512, activation='relu'),
    Dropout(0.5),
    Dense(2, activation='softmax')
])
```

VGG-19모델 생성 부분으로 include_top을 False로 해줌으로써 최상의 layer를 제외하였다. Flatten를 통해 데이터를 일차원으로 변환하고 Dense Layer 와 Dropout Layer를 사용하였다. 과적합 방지를 위해 Dropout를 0.5로 설정하였으며 Dense와 ReLU 활성화 함수를 통해 복잡한 비선형성을 학습을 진행한다. 그리고 2개의 뉴런을 가진 Dense Layer를 추가하여 최종 클래스 분류를 수행한다.

2.4 소프트 보팅

```
vgg_weight = 0.9
bilstm_weight = 0.1
final_deepvoice_prob = (vgg_weight * vgg_deepvoice_similarity_percent + bilstm_weight * bilstm_deepvoice_prob_percent) / (vgg_weight + bilstm_weight)
final_normal_voice_prob = (vgg_weight * vgg_normal_voice_similarity_percent + bilstm_weight * bilstm_normal_voice_prob_percent) / (vgg_weight + bilstm_weight)
```

보팅 이전에 TensorFlow에서 제공하는 keras에 있는 predict을 이용해 입력 데이터들의 예측과 결과를 반환하여 확률을 추출한다. 현재 진행기준 BiLSTM 모델의 정확도가 낮기 때문에 정확도가 높은 VGG-19모델의 가중치를 0.9 낮은 BiLSTM 모델의 정확도를 0.1을 주어 최종 결과를 판별한다.

2.5 애플리케이션 기능

```

import 'package:flutter/material.dart';
import 'package:file_picker/file_picker.dart';
import 'package:http/http.dart' as http;
import 'dart:convert';
  
```

material.dart: Flutter: 앱의 사용자 인터페이스를 만들기 위한 핵심 패키지이다. 아이콘과 같은 기능을 사용할 수 있다.

file_picker.dart: 애플리케이션기능에 파일 선택과 같은 작업을 수행하는데 필요한 패키지이다.

http.dart: Flask 애플리케이션에서 HTTP 요청과 응답을 다루기 위해 필요한 패키지이다.

dart:convert: 데이터를 인코딩하고 디코딩하는 데 사용되는 내장 라이브러리

3) 정확성

3.1 딥러닝 모델 정확성

1750/1750 [=====] - 37s 19ms/step
정확도: 0.5328212914173439

그림 14 BiLSTM모델 정확성

Found 204 images belonging to 2 classes.
7/7 [=====] - 2s 271ms/step - loss: 0.1364 - accuracy: 0.951
0
Test Accuracy: 0.9509803652763367

그림 15 VGG-19모델 정확성

남/여 같이 학습한 모델의 경우 BiLSTM의 같은 경우 53%의 매우 낮은 정확도를 보였다. 그러나 VGG-19모델의 경우에는 95%로 높은 정확도를 보였다. 그래서 소프트 보팅에 적용될 확률의 가중치를 VGG-19의 경우 0.9 BiLSTM의 경우 0.1을 적용하였다.

3.2 성별 분리 딥러닝 모델 정확도

남자/여자
정확도: 0.5328212914173439

→

남자
정확도: 0.5309580423508266

여자
정확도: 0.5248998282770464

BiLSTM의 경우 100개의 직접 녹음한 일반 음성과 100개의 학습에 사용하지 않은 딥보이스 음성을 만들어 정확도를 판별한 결과 정확도 향상을 보이지 않았다.

남자/여자
Test Accuracy: 0.9509803652763367

→

남자
Test Accuracy: 0.9801980257034302

여자
Test Accuracy: 0.9752475023269653

그림 17 VGG-19 모델 성별 분리 정확도

VGG-19모델의 경우에는 남자/여자 분리하여 딥러닝 모델을 생성했을 때 눈에 띄는 정확도 향상을 보였다. 그래서 VGG-19 모델만 여자, 남자 분리하여 모델을 생성하였다.

3.3 성별 분리 소프트 보팅 정확도

성별을 입력하세요 (여자/남자): 남자 1/1 [=====] - 0s 115ms/step 6/6 [=====] - 1s 39ms/step 최종 결과: 딥보이스 1/1 [=====] - 0s 107ms/step 8/8 [=====] - 0s 42ms/step 최종 결과: 딥보이스 1/1 [=====] - 0s 115ms/step 7/7 [=====] - 0s 31ms/step 최종 결과: 딥보이스 1/1 [=====] - 0s 115ms/step 16/16 [=====] - 0s 27ms/step	성별을 입력하세요 (여자/남자): 남자 1/1 [=====] - 0s 112ms/step 13/13 [=====] - 1s 25ms/step 최종 결과: 일반 음성 1/1 [=====] - 0s 105ms/step 10/10 [=====] - 0s 26ms/step 최종 결과: 일반 음성 1/1 [=====] - 0s 116ms/step 18/18 [=====] - 1s 32ms/step 최종 결과: 일반 음성 1/1 [=====] - 0s 109ms/step 9/9 [=====] - 0s 34ms/step 최종 결과: 일반 음성 1/1 [=====] - 0s 109ms/step
--	--

소프트 보팅을 적용한 코드에 남자 모델의 정확도는 총 200의 음성 판별 시에 4개의 오답으로 98%의 정확도를 보였다.

성별을 입력하세요 (여자/남자): 여자 1/1 [=====] - 19s 19s/step 7/7 [=====] - 8s 25ms/step 최종 결과: 답보이스 1/1 [=====] - 0s 113ms/step 10/10 [=====] - 0s 15ms/step 최종 결과: 답보이스 1/1 [=====] - 0s 111ms/step 10/10 [=====] - 0s 15ms/step	성별을 입력하세요 (여자/남자): 여자 1/1 [=====] - 0s 115ms/step 15/15 [=====] - 1s 20ms/step 최종 결과: 일반 음성 1/1 [=====] - 0s 120ms/step 13/13 [=====] - 0s 14ms/step 최종 결과: 일반 음성 1/1 [=====] - 0s 107ms/step
---	--

소프트 보팅을 적용한 코드에 여자 모델의 정확도는 200의 음성 판별 시에 5개의 오답으로 97.5%의 정확도를 보였다. 그래서 소프트 보팅 코드의 정확도가 남/여 일 때 95%인거에 비하여 남과 여를 분리 하였을 때 성능향상을 보였다.

<< 디자인/제작물 >>

1) 관련 시안



그림 20 시안 메인
인터페이스 화면



그림 21 시안 결과 출력 인터페이스

애플리케이션 메인 화면에 사용자가 판별음성을 업로드 할 수 있는 UI와 성별을 선택할 수 있는 버튼을 제공한다. 그리고 서버로 전송을 하는 판별시작 버튼을 제공한다. 그리고 서버에서 판별 결과에 따른 인터페이스를 출력한다.

2) 최종 제작물

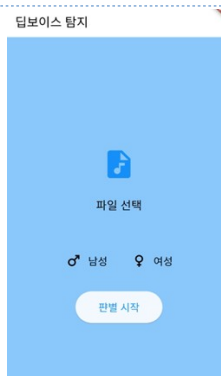


그림 22 메인
인터페이스

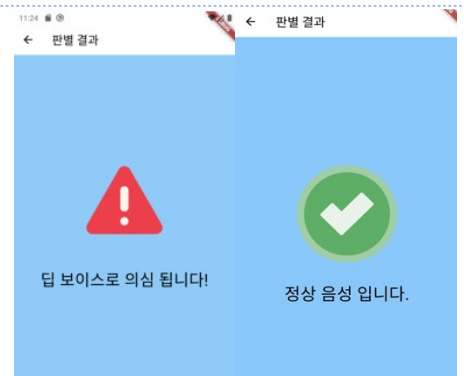


그림 23 결과 출력 인터페이스

애플리케이션 file_picker를 통해 사용자는 휴대폰에서 판별한 음성을 업로드 할 수 있으면 성별 선택 아이콘을 통해 성별을 할 수 있다. 판별 시작을 하면 flask 서버에 전송해 성별 정보에 맞는 모델로 판별하여 판별 결과를 애플리케이션 전송하고 해당하는 인터페이스가 출력된다.

6. 보안 및 활용 계획


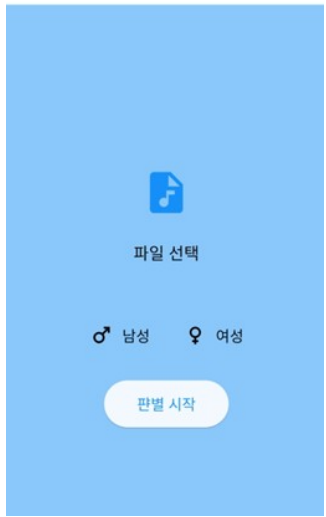
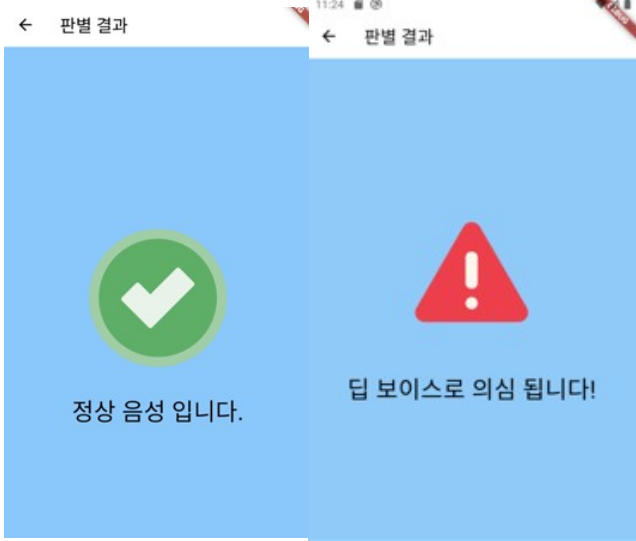
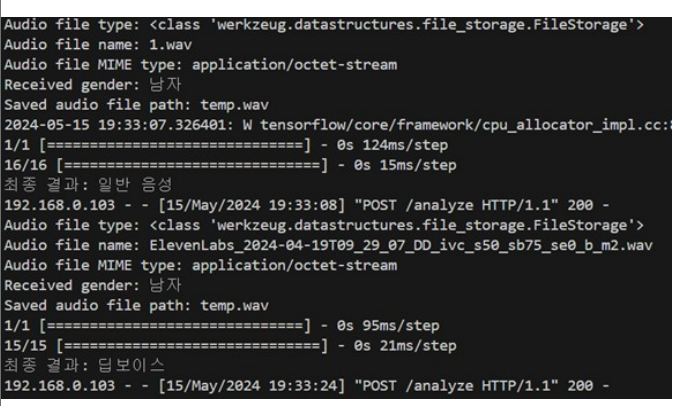
1) 향후 보안 사항

학습 음성 데이터가 충분한 확보가 된다면 카테고리를 성별뿐 아니라 나이, 지역등 다양한 카테고리 제공한 모델을 만들어 더 정확한 판별이 가능하게 보안이 가능하다. 그리고 VGG-19의 정확도에 비해 아쉬운 BiLSTM모델의 정확도를 올려 가중치 값을 똑같이 주어 더 정확성을 올릴 수 있을 것이다. 애플리케이션에는 녹음한 음성을 가져와서 판별하는 것이 아닌 통화를 할 때 자동으로 애플리케이션과 연동이 된다면 사용자가 좀 더 편의성 있게 사용가능할 것이라 생각한다.

2) 전시/출품 계획

현재 해당 프로젝트를 기반으로 교육부에서 주최하는 2024 학생 창업유망팀 u300경진대회에 참여하여 심사중에 있다. 그리고 앱 스토어와 구글 플레이에 딥보이스 판별관련 애플리케이션 검색진행시 관련 어플의 접근성이 힘든 것을 확인하였다. 그래서 우리의 애플리케이션을 출시한다면 딥보이스를 이용한 보이스 피싱 사례가 줄어들 거라 생각한다. 그리고 일반 사용자가 아니더라도 군사 기지에 음성 인증 시스템을 구축하여 군인들의 신원을 확인 할 수 있는 군용 과 음성 특징 추출 기술을 사용해 환자의 목소리를 분석하여 의료 진단이나 치료에 도움이 될 수 있는 의료 분야에도 출품 가능할 것이라 생각한다.

7. 실적물 사진

 <p>gbestmodel mbestmodel</p>	<p>딥보이스 탐지</p> 
<p>그림 24 h5 형식의 남/여 모델 생성</p> 	 <pre> Audio file type: <class 'werkzeug.datastructures.file_storage.FileStorage'> Audio file name: 1.wav Audio file MIME type: application/octet-stream Received gender: 남자 Saved audio file path: temp.wav 2024-05-15 19:33:07.326401: W tensorflow/core/framework/cpu_allocator_impl.cc: 1/1 [=====] - 0s 124ms/step 16/16 [=====] - 0s 15ms/step 최종 결과: 일반 음성 192.168.0.103 - - [15/May/2024 19:33:08] "POST /analyze HTTP/1.1" 200 - Audio file type: <class 'werkzeug.datastructures.file_storage.FileStorage'> Audio file name: ElevenLabs_2024-04-19T09_29_07_DD_ivc_s50_sb75_se0_b_m2.wav Audio file MIME type: application/octet-stream Received gender: 남자 Saved audio file path: temp.wav 1/1 [=====] - 0s 95ms/step 15/15 [=====] - 0s 21ms/step 최종 결과: 딥보이스 192.168.0.103 - - [15/May/2024 19:33:24] "POST /analyze HTTP/1.1" 200 - </pre> <p>그림 27 flask 서버에서의 정상 동작</p>