AI Speech Supporter

AISS팀

김동한
김준서

CONTENTS

1

문제점 정의

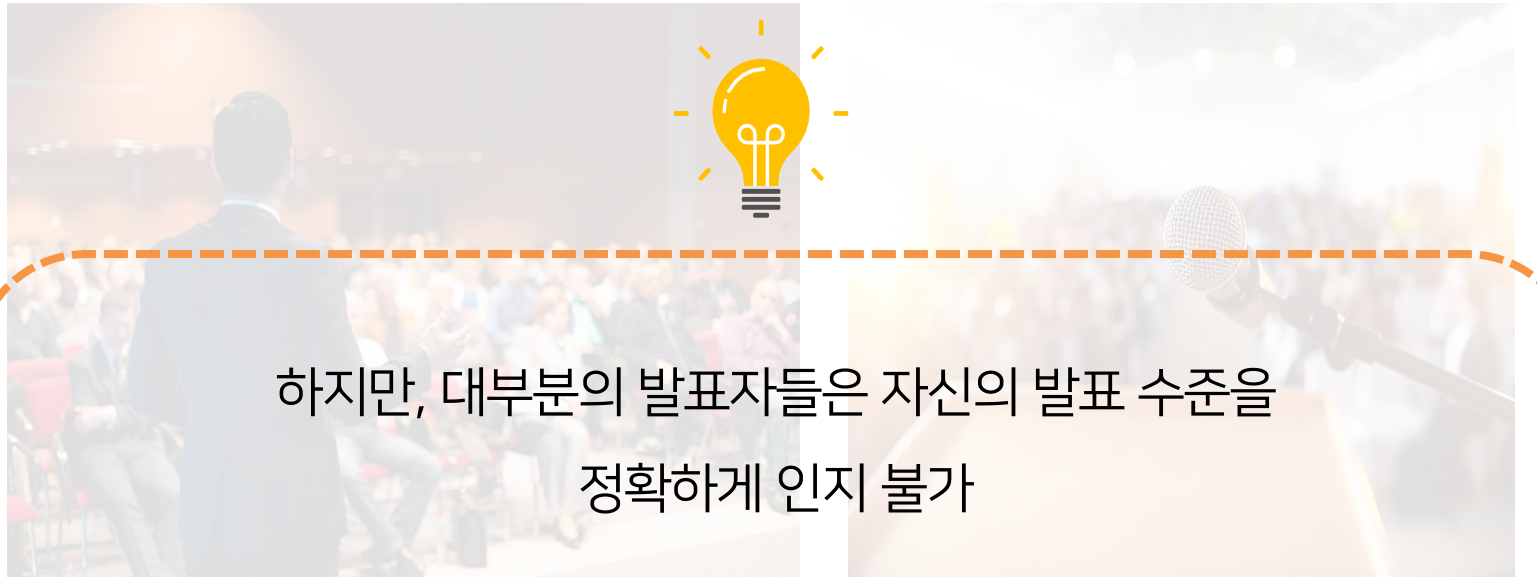
1 문제점 정의

다양한 환경에서 직면하는 발표

상대방에게 정보 제공이나, 자신의 의견을 발표하기 위해서
발표를 상황에 직면하는 경우가 증가

1 문제점 정의

다양한 환경에서 직면하는 발표

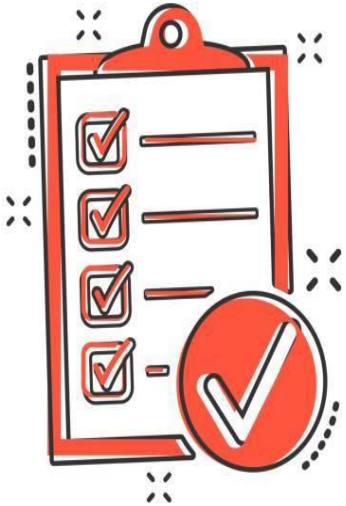


하지만, 대부분의 발표자들은 자신의 발표 수준을
정확하게 인지 불가

상대방에게 정보 제공이나, 자신의 의견을 발표하기 위해서
발표를 상황에 직면하는 경우가 증가

1 문제점 정의

자가 진단의 어려움



경험적으로 여러 매체를 통해서 발표 연습을 진행하더라도

자신의 발표가 어땠는지, 어느 부분이 부족한지 판단 어려움

2

발표 종합 진단 서비스: AISS

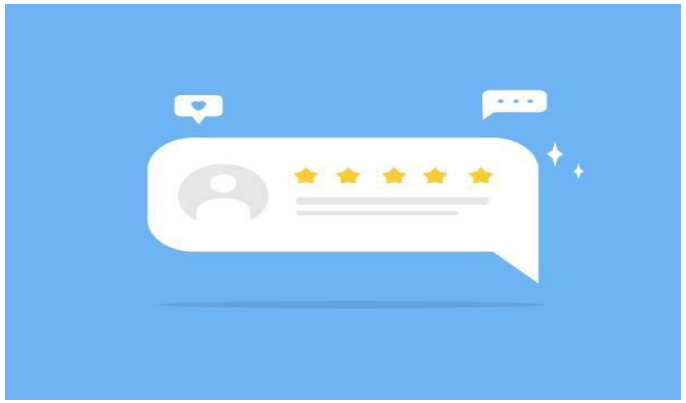
2 발표 종합 진단 서비스: AISS

발표 종합 진단 서비스

AI Speech Suppoter(AISS)

사용자의 발표 영상에서 텍스트, 음성, 비디오에서 특징 추출

멀티모달을 통해서 좋은 발표 여부를 판단해주는 발표 종합 진단 서비스



전체적인 발표 평가를 통해서 좋은 발표 여부를 진단

특정 기준에 따른 발표에 대한 평가를 텍스트로 전달

2 발표 종합 진단 서비스: AISS

서비스 주요 기능

주요 기능

발표 영상 촬영

앱 내에서 촬영 기능 제공, 사전에 설정된 각도에 따라 촬영을 진행할 수 있도록 가이드라인을 제공

발표 능력 진단

발표 영상에서 각 특징들을 분석

분석 결과 진단서 제공

분석한 결과를 바탕으로 종합적인 발표 능력 진단서를 제공

피드백 제공

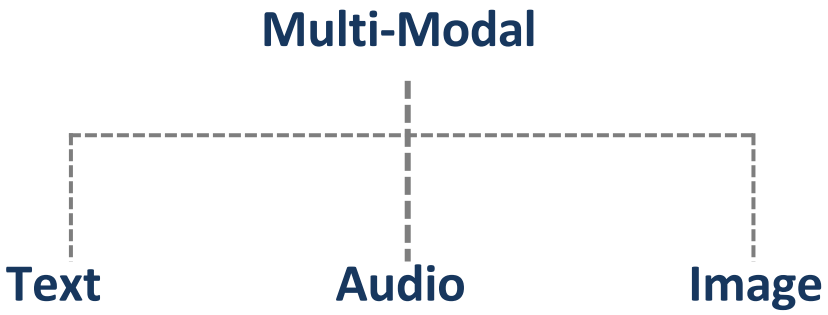
사용자의 취약점에 알맞는 피드백을 제공, 발표 능력을 개선할 수 있도록 조언

발표 영상을 촬영 이후에, 각 특징들을 통해서 분석,
분석 결과를 바탕으로 종합적인 발표 능력 진단서 제공

2 발표 종합 진단 서비스: AISS

진단 서비스 Flow

발표 종합 진단 서비스



Uni-Modal에 있어서는

각 데이터 특징에 따라 적절한 모델 사용

다양한 특징을 지는 데이터에서 feature을 추출

각각의 특성을 가진 특징을 결합하여 발표 영상 평가

2 발표 종합 진단 서비스: AISS

진단 서비스 Flow

발표 종합 진단 서비스

Multi-Modal

데이터의 특성을 추출하기 위해서

각 데이터에 적합한 모델을 선정

Uni-Modal에 있어서는

각 데이터 특징에 따라 적절한 모델 사용

다양한 특징을 지는 데이터에서 feature을 추출

각각의 특성을 가진 특징을 결합하여 발표 영상 평가

3

활용 기술 설명

활용 기술 설명

Dataset

Good Dataset VS Bad dataset

Good Presentation Dataset → TEDLIUM

Poor Presentation Dataset → MOSI

3 활용 기술 설명

Dataset

Good Dataset

TED(Technology, Entertainment, Design) 강연 영상 데이터셋

뛰어난 발표 영상 데이터를 제공, 총 777시간의 오디오 데이터셋 → 4620개의 발표 제공

활용 기술 설명

Dataset

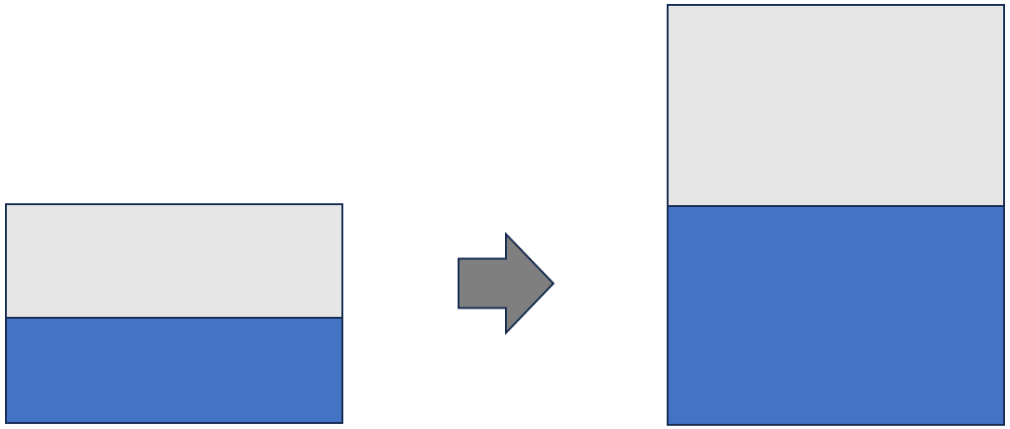
Bad dataset

MOSI(Multimodal Corpus of Sentiment Intensity)데이터셋은 발표자가 실수하거나 감정적으로 불안정할 때 발생하는 다양한 오류와 잘못된 발표를 포함 → 92개의 발표 제공

활용 기술 설명

Dataset

Bad dataset의 문제점



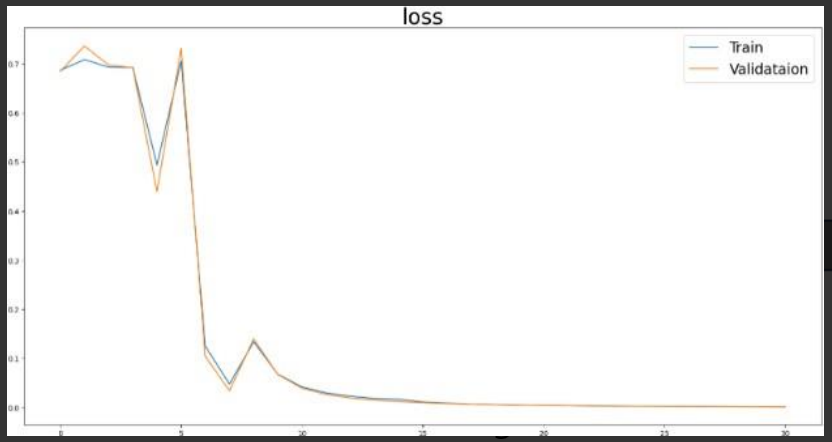
데이터를 두배만큼 복제하여 학습에 사용

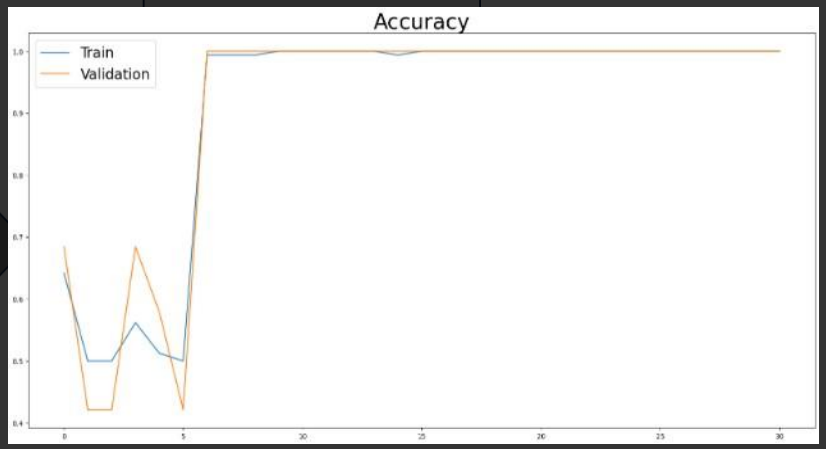
동일한 데이터 학습으로 인해서 과적합 발생 가능하나, 진동을 어느정도 잡아줄 것이라고 생각

3 활용 기술 설명

Dataset

Bad dataset의 문제점





여전히 신뢰할만한 학습 성능을 보여주지 못했다.

동일한 데이터 학[illegible] 것이라고 생각

추후, poo[illegible] 수정해볼 예정

저번 발표에서 교수님께서 주신 피드백을 통해서 습득한

나쁜 데이터를 만들어보는 것이 어떻겠냐는 피드백 수용

활용 기술 설명

Dataset

Bad dataset의 해결 방안

현재 네트웍개론과 같은 수업에서 영어 기반 발표 수업을 진행중

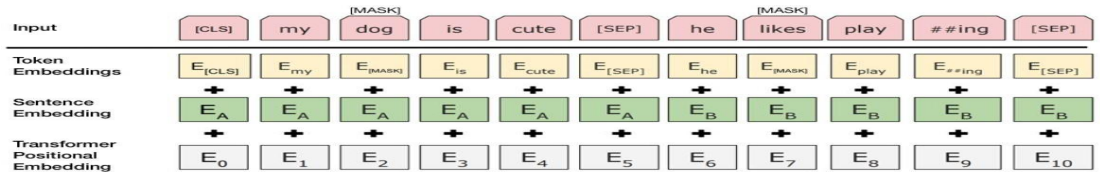
다음과 같은 발표 수업에서 발표 연습할 때마다 발표 영상을 촬영

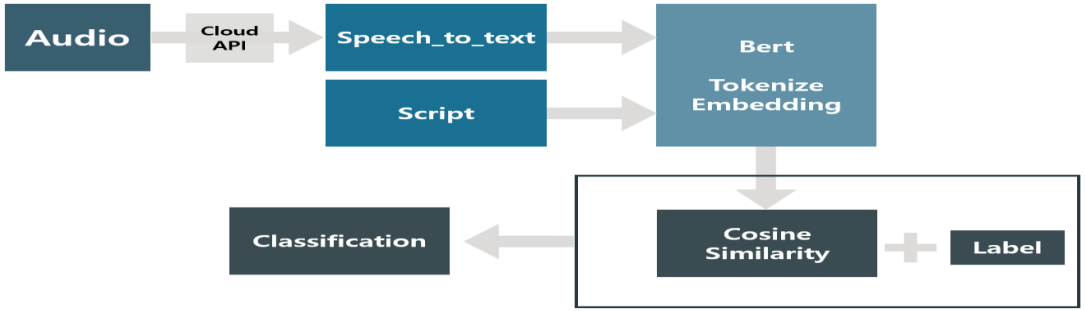
추가적으로 수업 내 타 팀원들의 발표 영상 또한 제공받음

활용 기술 설명

Text Model: Cosine Similarity

클러스터링



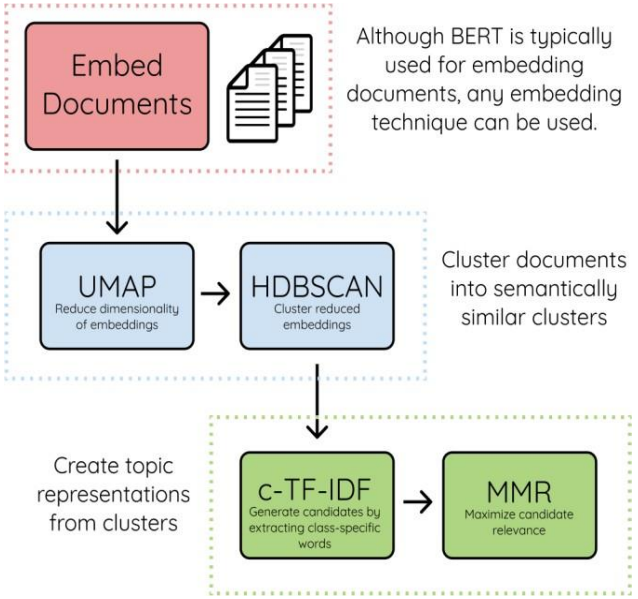


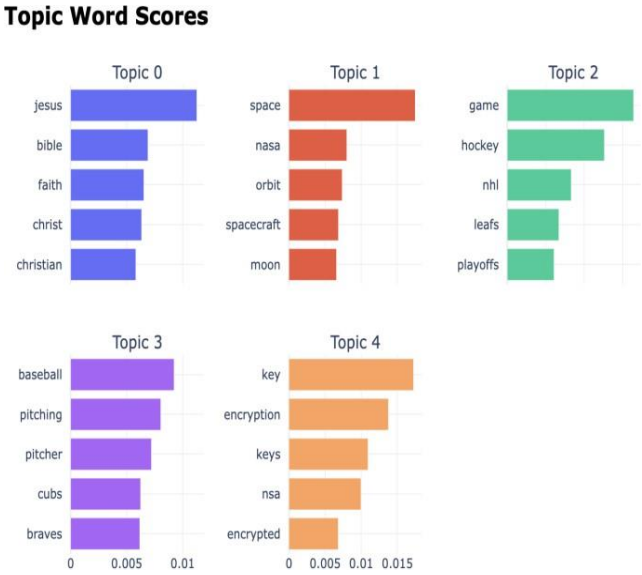
다음과 같은 구조를 통해서 대본과 Speech-to-text API에서 얻은
데이터 간의 코사인 유사도를 측정
→ 대본과 Speech-to-text API에서 얻은 데이터 간의 정확도 측정

활용 기술 설명

Text Model: BERTopic

토픽 유사도 측정 모델





BERTopic을 통해서,

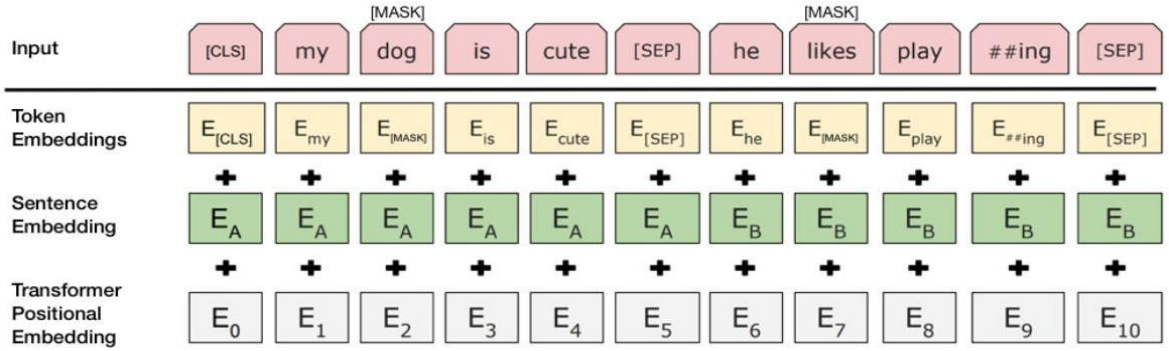
한 발표 내의 문장들 간의 토픽 유사도를 구하고 하나의 변수로 생성

Text Model: Next Sentence Probability

발표 문장 순서 적절성 측정 모델

나는 밥을 먹는다. → 나는 식당에 왔다.

나는 식당을 왔다. → 나는 밥을 먹는다.



두 문장이 비슷한 주제를 시사한다고 하더라도

그 순서에 따라서 의미와 이해가 달라지게 됨

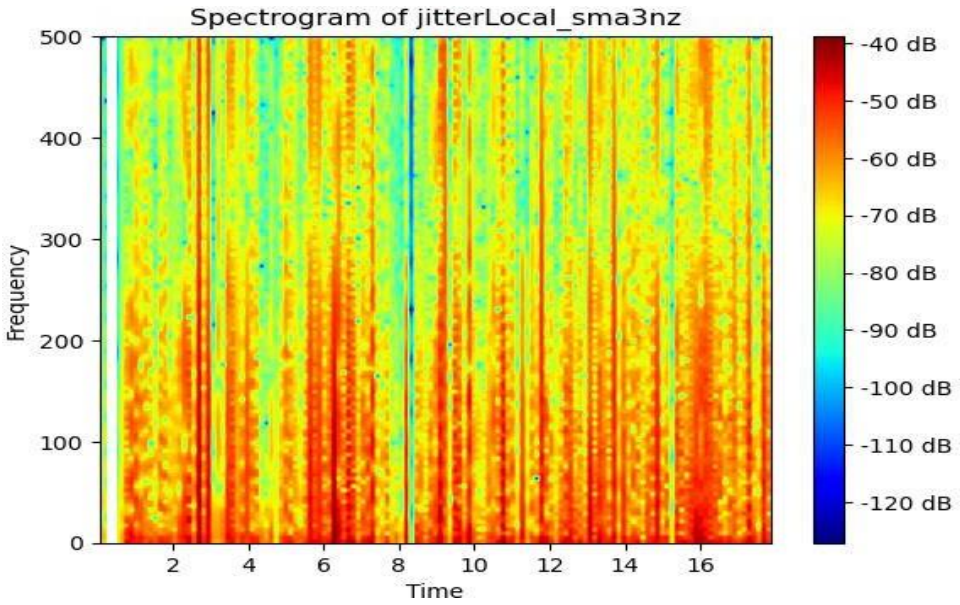
NSP는 주어진 두 개의 문장이 실제로 연속적으로 이어지는지 여부를 예측하는 작업을 의미

→ 적절히 문장이 따라오고 있는가에 대한 확률 추가

활용 기술 설명

Audio Model: jitterLocal feature using openSMILE API

발성 안정성 측정 모델



openSMILE API: 입력 받은 음성으로부터 여러가지 특징 추출

jitterLocal_sma3nz: 음성의 세기 변동에 대한 주파수 변동 비율을 나타내는 특징

→ 음성의 주파수 변동을 측정하여 발성의 안정성을 추정

활용 기술 설명

Vision Model: L2CS - Net

시선 불안정성 측정 모델

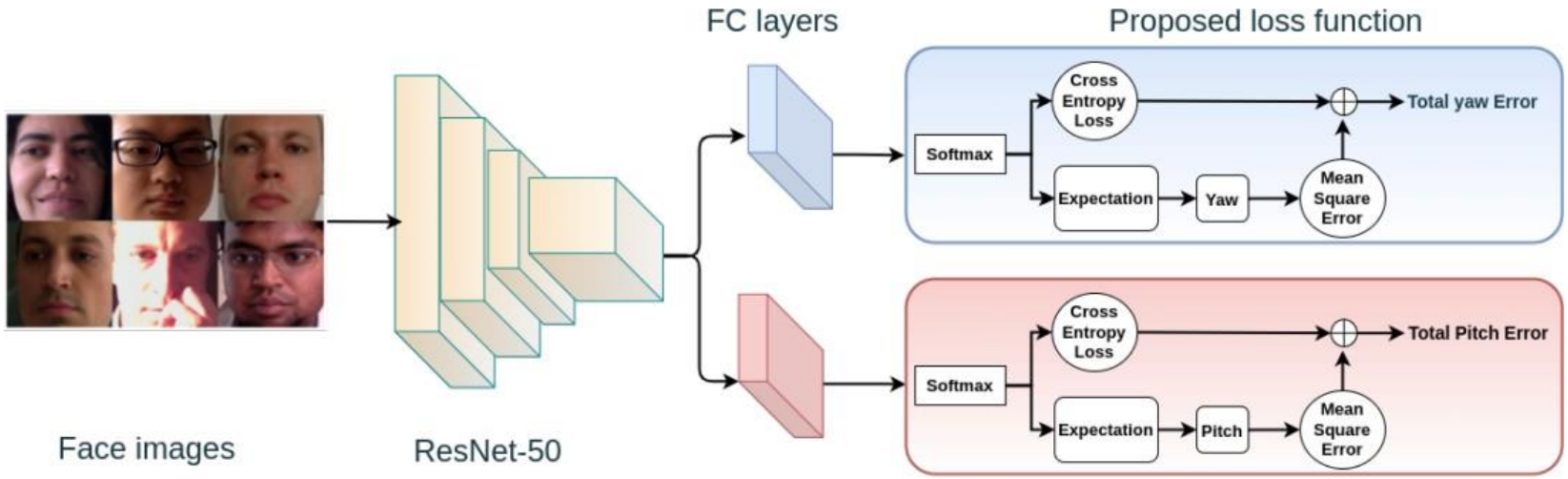


Fig. 1: L2CS-Net with combined classification and regression losses.

횡적 종적 시선 각도(yaw, pitch)를 개별적으로 회귀를 통해 측정

이를 통해서, 어느 특정 지점을 바라보고 있는지 측정 가능

각도 변화를 측정 → 눈동자 움직임 정도 측정 가능

활용 기술 설명

Vision Model: L2CS - Net

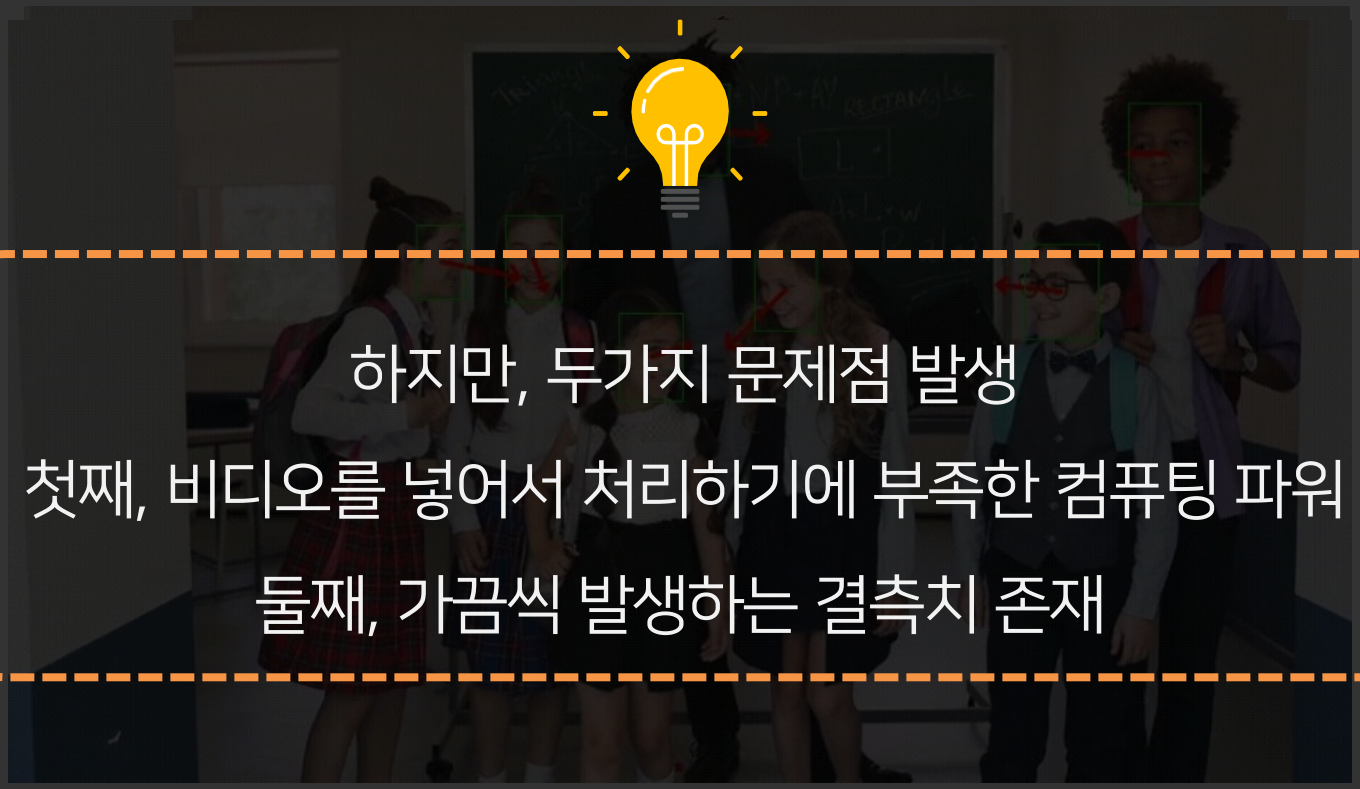
시선 불안정성 측정 모델

다음과 같이 각 대상들의 시선이 어디를 바라보고 있는지 측정이 가능

3 활용 기술 설명

Vision Model: L2CS - Net

시선 불안정성 측정 모델



하지만, 두가지 문제점 발생

첫째, 비디오를 넣어서 처리하기에 부족한 컴퓨팅 파워

둘째, 가끔씩 발생하는 결측치 존재

다음과 같이 각 대상들의 시선이 어디를 바라보고 있는지 측정이 가능

Vision Model: L2CS - Net

시선 불안정성 측정 모델

eUclidean Distance

$$\text{eUclidean Dist.} = (x_2 - x_1)^2 + (y_2 - y_1)^2$$

L2CS-net를 통해 pitch_predicted_list(횡적),

Yaw_predicted_list(종적) 포인트 좌표 측정

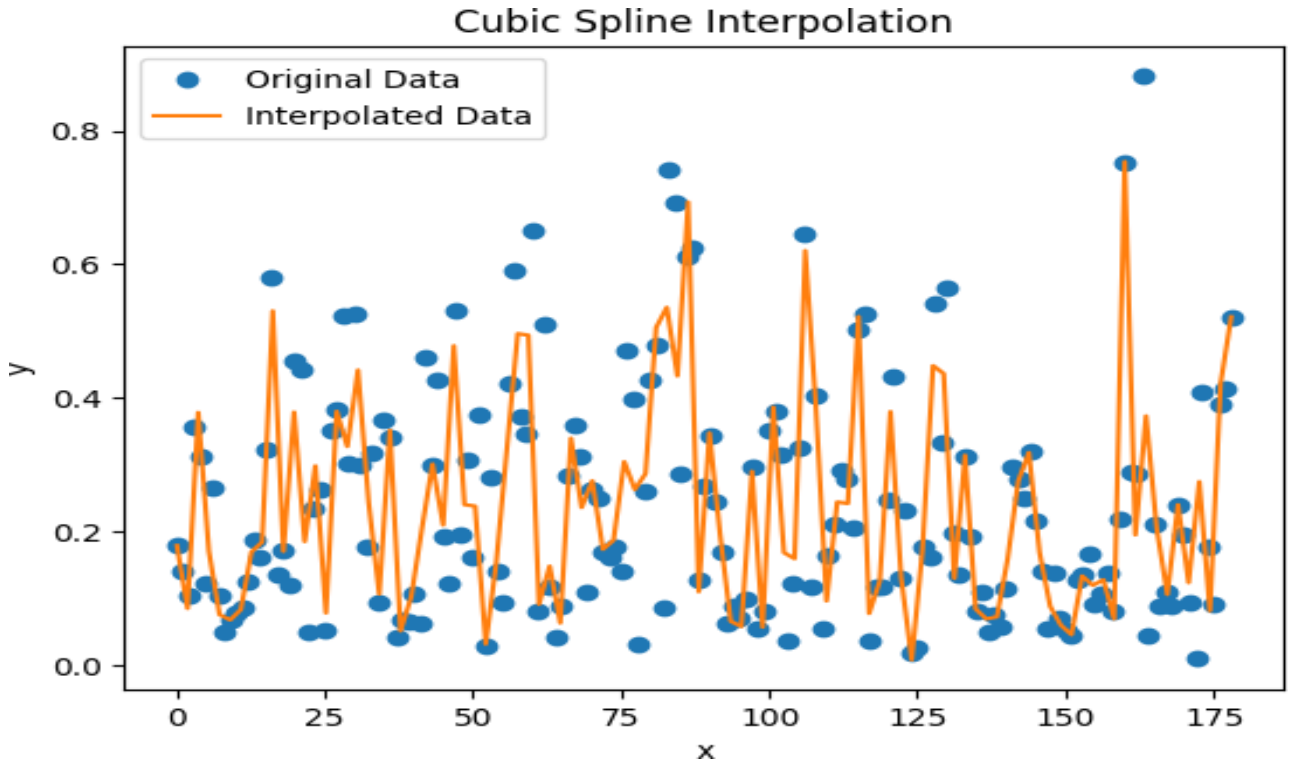
∨

eUclidean Distance 통해서 시선 이동 거리 측정

활용 기술 설명

Vision Model: L2CS - Net

시선 불안정성 측정 모델



시선의 추세를 반영하면 보간을 진행할 수 있을 것이라 생각

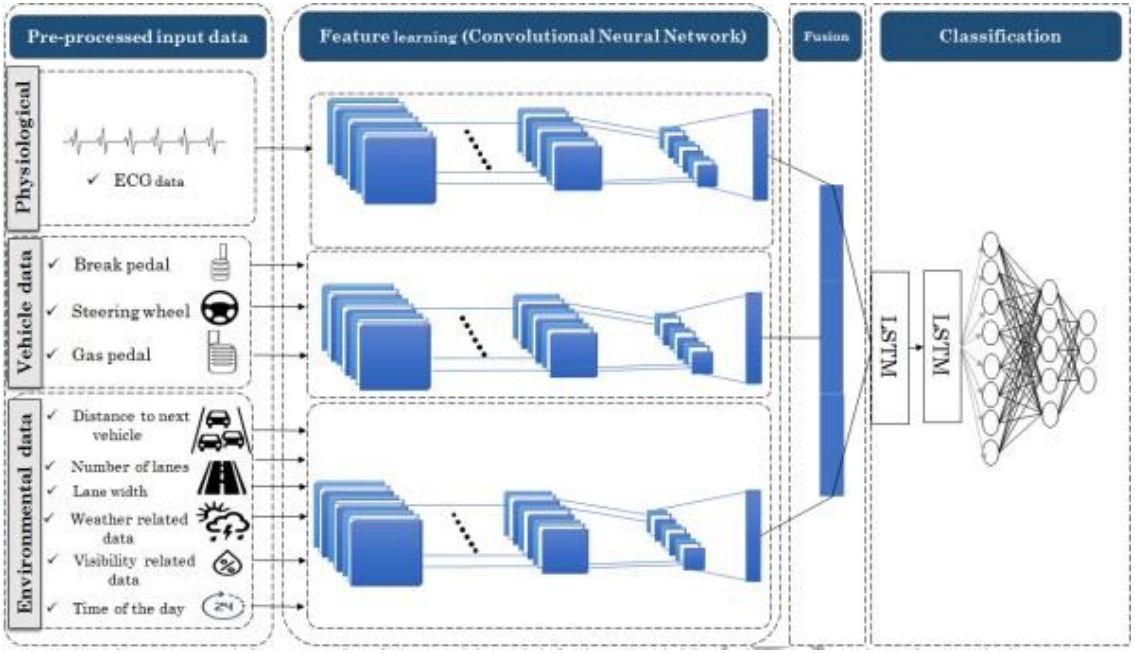
Cubic Spline Model을 통해서 보간 진행

활용 기술 설명

Multi Modal: Feature vector Concatenation

특징을 1차원 벡터로 변환, 이어 붙여서 함께 학습

"Automatic Driver Stress Level Classification Using Multimodal Deep Learning" by Mohammad Naim Rastgoo



Feature Vector를 하나로 이어 붙여서 1차원 백터로 변환, 이를 LSTM을 사용하여 멀티모달 진행하는 방법 사용

활용 기술 설명

Multi Modal: Feature vector Concatenation

특징을 1차원 벡터로 변환, 이어 붙여서 함께 학습

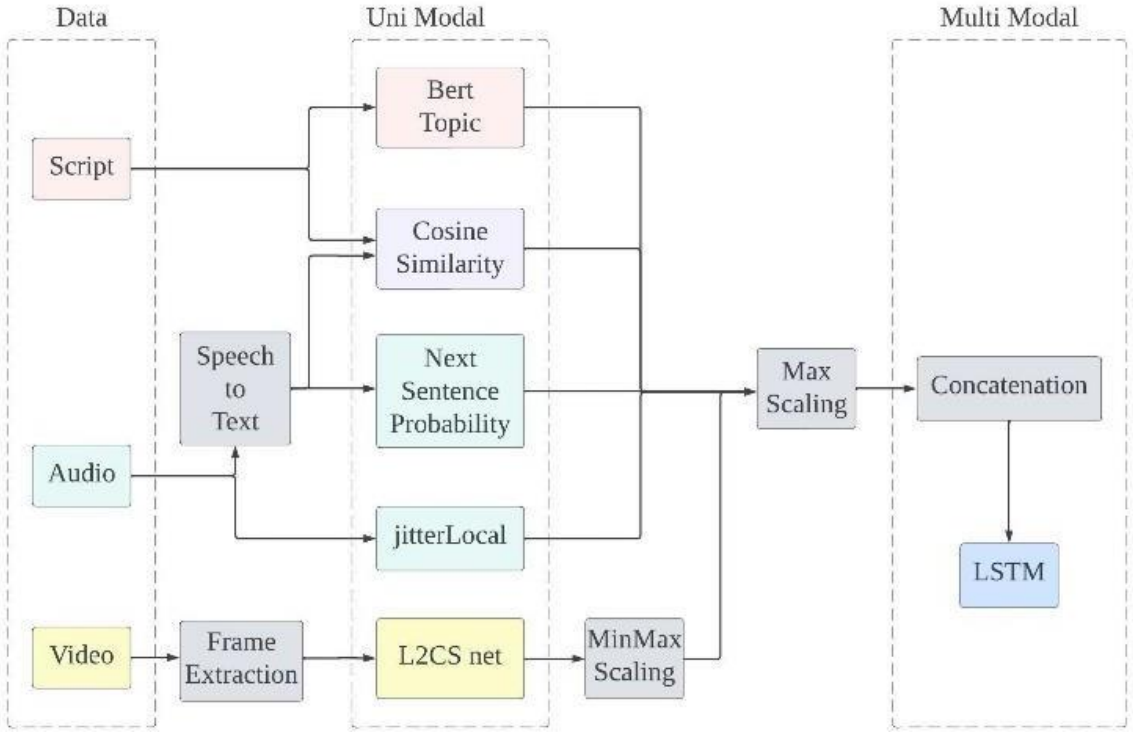
Kind	length
Bert Topic Feature Vector	1
Cosine Similarity	0.057
Next Sentence Probability	1
L2CS net	0.069
jitterLocal feature	0.000736

Feature vector Max Scaling:
Feature vector의 길이 별로 가중치 적용(Max Scaling)
→ 길이가 긴 Feature일수록 작은 값을 곱하여 각 Feature의 영향력을 동일하게

활용 기술 설명

Multi Modal: Feature vector Concatenation

특징을 1차원 벡터로 변환, 이어 붙여서 함께 학습



Feature Vector Concatenation을 통해서 하나의 1차원 백터로 생성 선행 연구와 동일하게 LSTM 모델을 통해서 Fusion 진행

활용 기술 설명

Multi Modal: Feature vector Concatenation

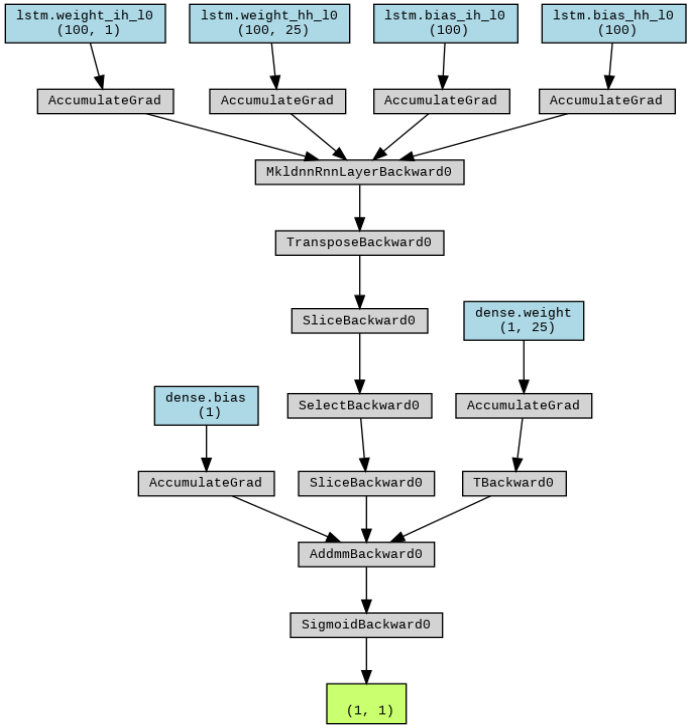
특징을 1차원 벡터로 변환, 이어 붙여서 함께 학습

```
import torch.nn as nn
# LSTM model definition
class LSTM(nn.Module):
    def __init__(self, input_size, hidden_size):
        super(LSTM, self).__init__()
        self.lstm = nn.LSTM(input_size, hidden_size, batch_first=True)
        self.dense = nn.Linear(in_features=hidden_size, out_features=1)

    def forward(self, x):
        output, _ = self.lstm(x)
        output = self.dense(output[:, -1, :])
        output = torch.sigmoid(output)
        return output

input_size = 1  # 입력 텐서의 feature 차원 크기
hidden_size = 25
num_epochs = 50

model = LSTM(input_size, hidden_size, output_size)
```



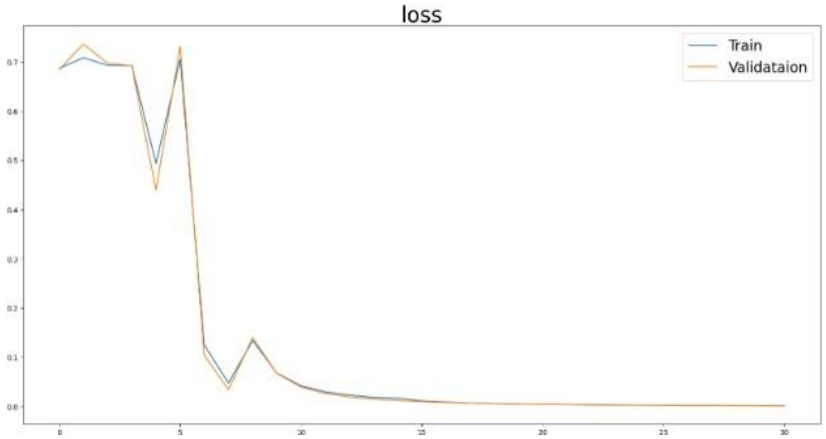
Feature Vector Concatenation을 통해서 하나의 1차원 백터로 생성 선행 연구와 동일하게 LSTM 모델을 통해서 Fusion 진행

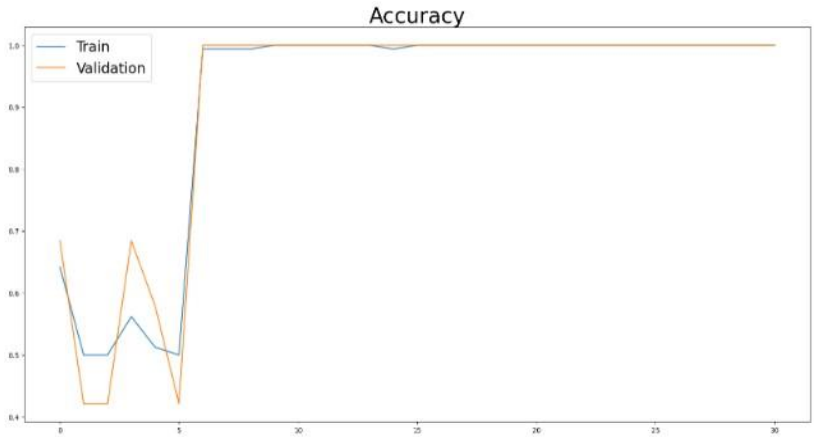
활용 기술 설명

Multi Modal: Feature vector Concatenation

특징을 1차원 벡터로 변환, 이어 붙여서 함께 학습

Step2: #Hidden State = 25, #Data = 262



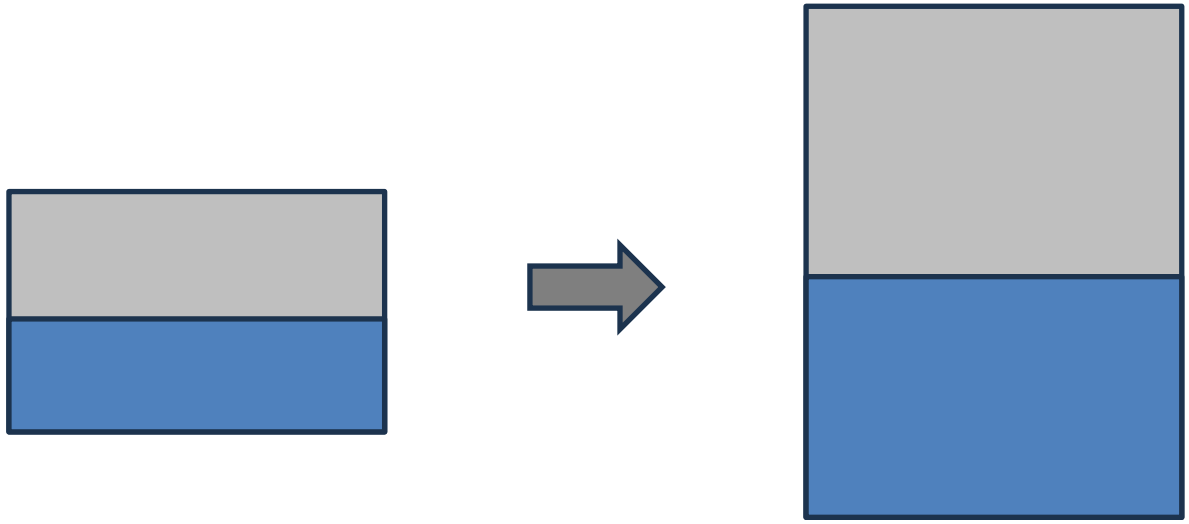


이상적이진 않지만 진동이 존재하는 학습을 진행했던 저번 과제
데이터 부족을 문제점으로 판단 → **데이터셋 추가!**

활용 기술 설명

Multi Modal: Adding More Dataset

Bad Dataset 추가 → **Good Dataset 까지 추가 가능**



bad data를 직접 제작하여, 262개로 증가

이후에 동일하게 Oversampling 해서 good data 524개, bad data 524개로 증강

동일한 데이터 학습으로 인해서 과적합 발생 가능하나, 이전보다는 적절한 결과 예상

활용 기술 설명

Multi Modal: Feature vector Concatenation

Dataset 증강 이후 모델 학습

Stacked LSTM: #Hidden State = 25, #Data =181

Stacked LSTM: #Hidden State = 25, #Data =1048

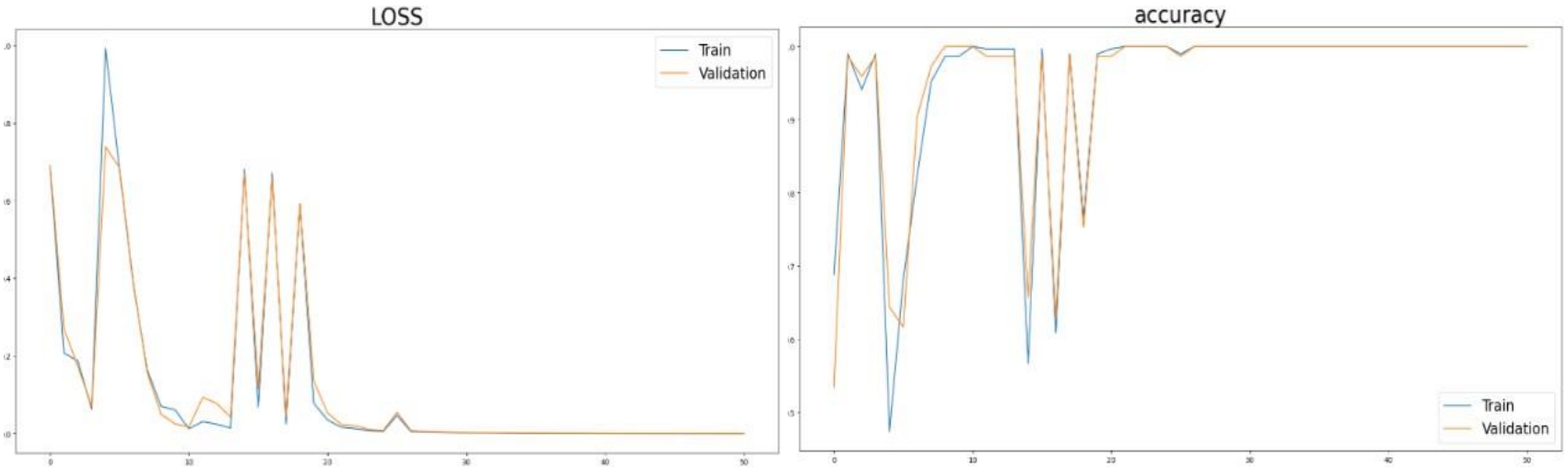
데이터수가 늘어난 만큼, 이전의 단일 LSTM에서 Stacked LSTM 모델을 사용
데이터 수가 증강된 만큼 이전보다는 진동이 감소할 것이라고 생각

활용 기술 설명

Multi Modal: Feature vector Concatenation

Dataset 증강 이후 모델 학습

데이터 증강 이전
#Data =181



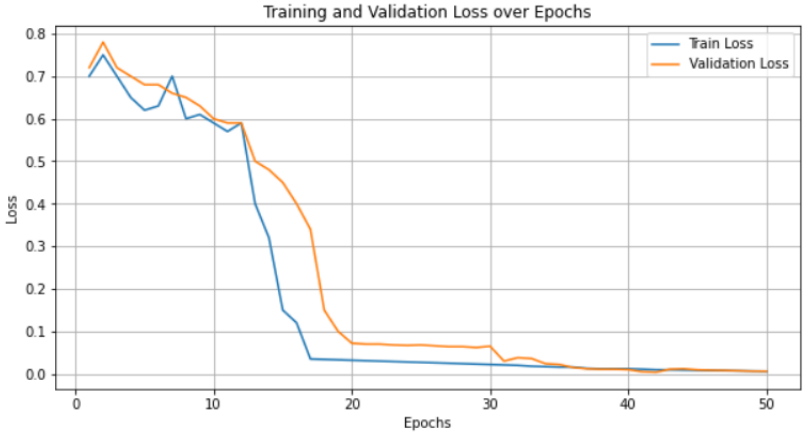
데이터 증강 이전의 Stacked LSTM모델을 사용할 경우에
데이터가 적은 문제로 인해서 진동하는 문제가 발생

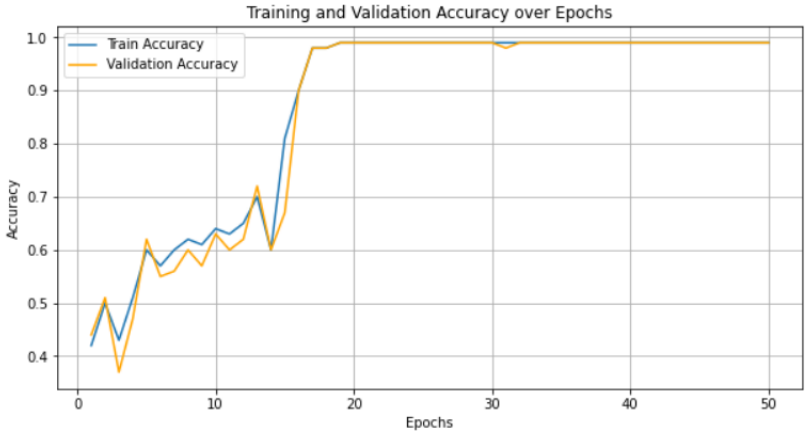
활용 기술 설명

Multi Modal: Feature vector Concatenation

Dataset 증강 이후 모델 학습

데이터 증강 이후
#Data =1048





여전히 진동이 발생하나, 이전에 비해서 감소 → 여전히 이상적인 그래프는 아님
모델적인 변화도 필요할 것이라고 판단됨 →**여전히 데이터에 비해서 모델이 복잡**

활용 기술 설명

Multi Modal: Feature vector Concatenation

Dataset 증강 이후 모델 학습

Test data	Real	Prediction	
1	1	1	T
2	1	1	T
3	0	1	F
…			
50	0	0	T

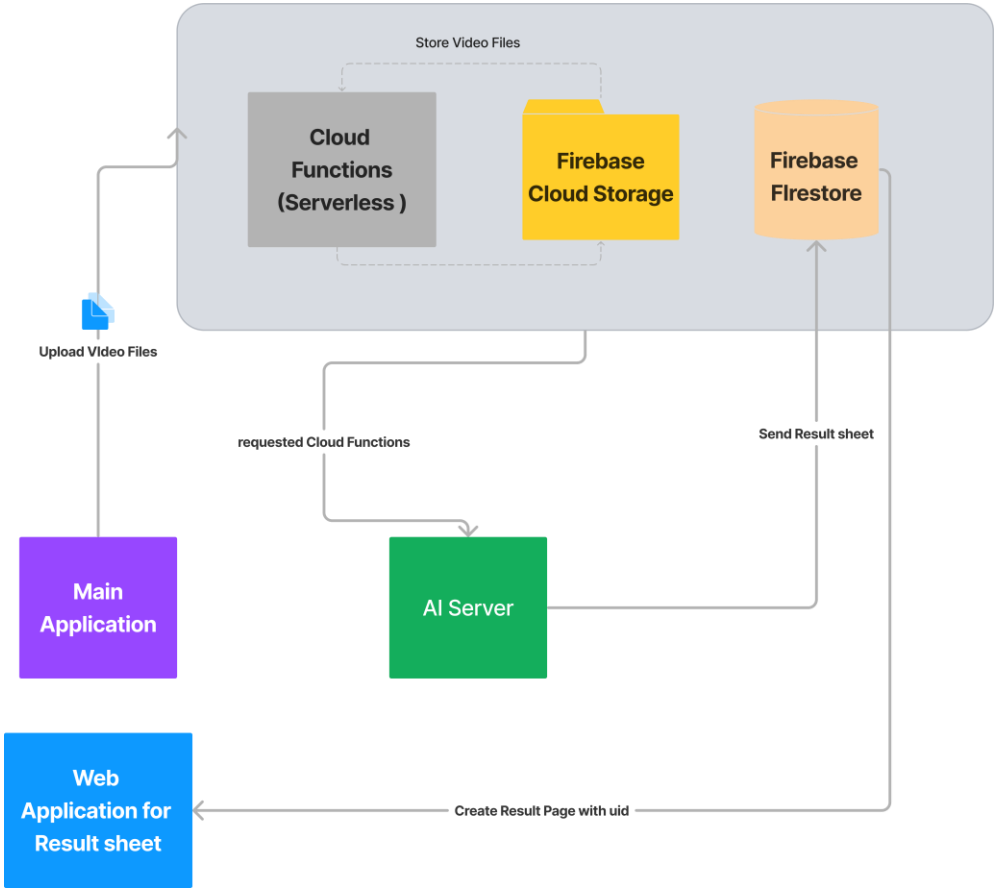
50개의 테스트를 진행했을 때, 3개의 데이터를 적절히 예측하지 못함
부적절한 데이터일 경우에도 특징이 명확하지 못하면 부정적이라 판단하기 어려움
또한 적절한 데이터일 경우에도, 화면 전환과 같이 예측하지 못하는 상황 발생 시 예측 실패

4

Application

4 Application 구현

Overall Structure



4 Application 구현

Process

1.사용자가 메인 어플리케이션에서 비디오 파일을 촬영 또는 업로드

2.결과 시트를 전달받을 이메일을 첨부하여 전송 실행

3.전송 성공 시 Firebase Storage 공간에 비디오 파일을 업로드

4.Cloud Functions를 이용하여 Storage에 새로운 비디오 파일이 생성되면 AI 서버로 연산 처리 요청

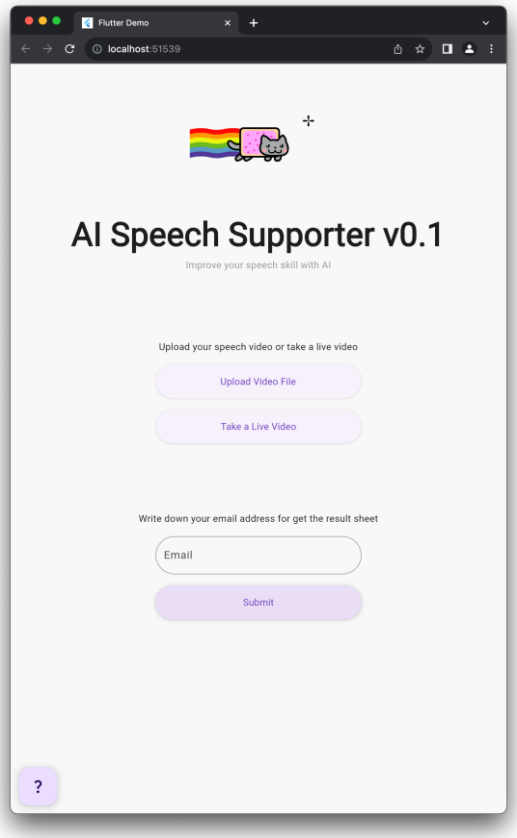
5.AI 서버로부터 연산 과정이 끝나면 Firebase Firestore 데이터베이스에 결과값을 저장

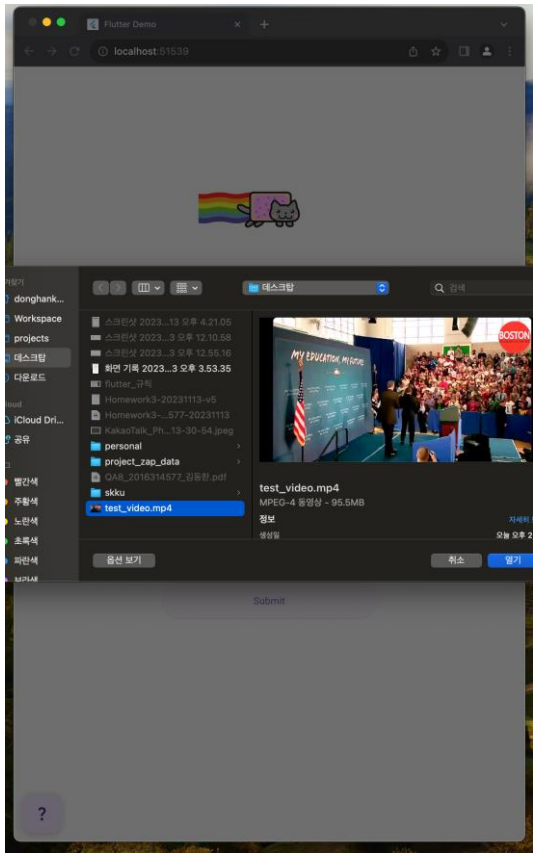
6.Cloud Functions에서 이벤트 트리거를 감지하여 새로운 값이 데이터베이스에 들어오면
해당 데이터를 조회

7.Naver SENS API를 활용하여 해당 유저의 이메일로 결과 정보를 담고 있는 데이터의 uid값을 포함한
결과 시트 확인용 페이지 URL을 생성 후 전송

4 Application 구현

Process

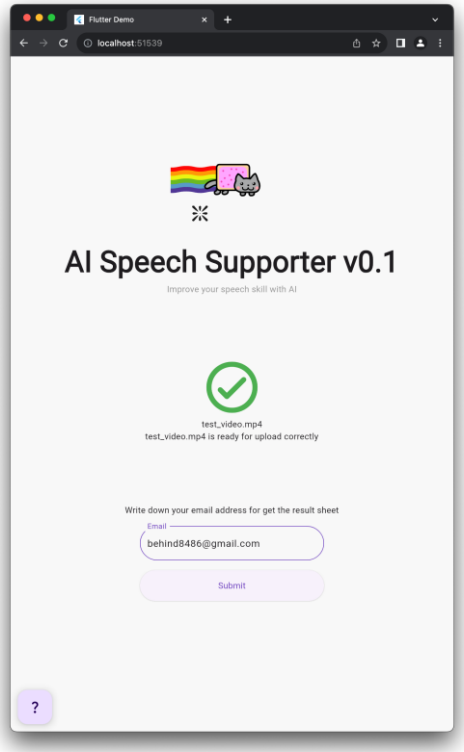




사용자가 메인 어플리케이션에서 비디오 파일을 촬영 또는 업로드

4 Application 구현

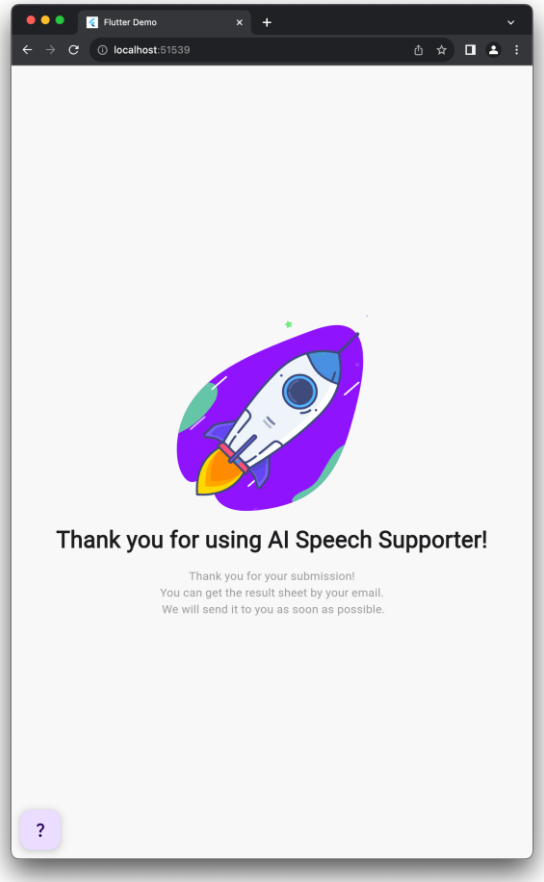
Process

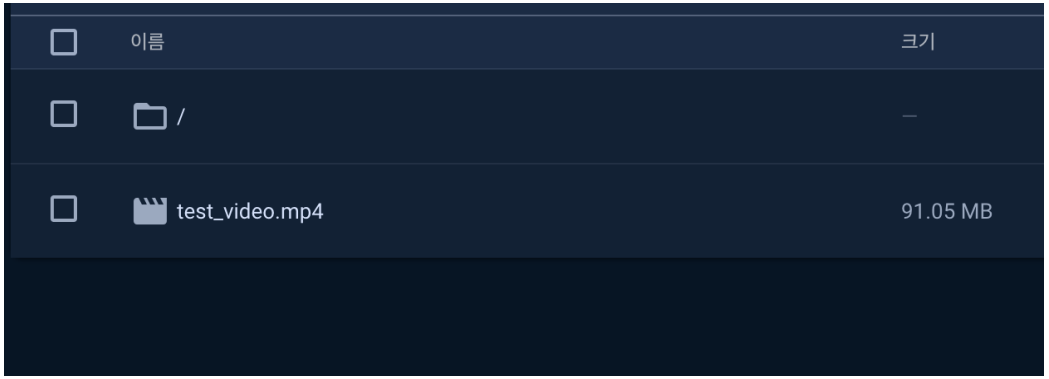


결과 시트를 전달받을 이메일을 첨부하여 전송 실행

4 Application 구현

Process





전송 성공 시 Firebase Storage 공간에 비디오 파일을 업로드

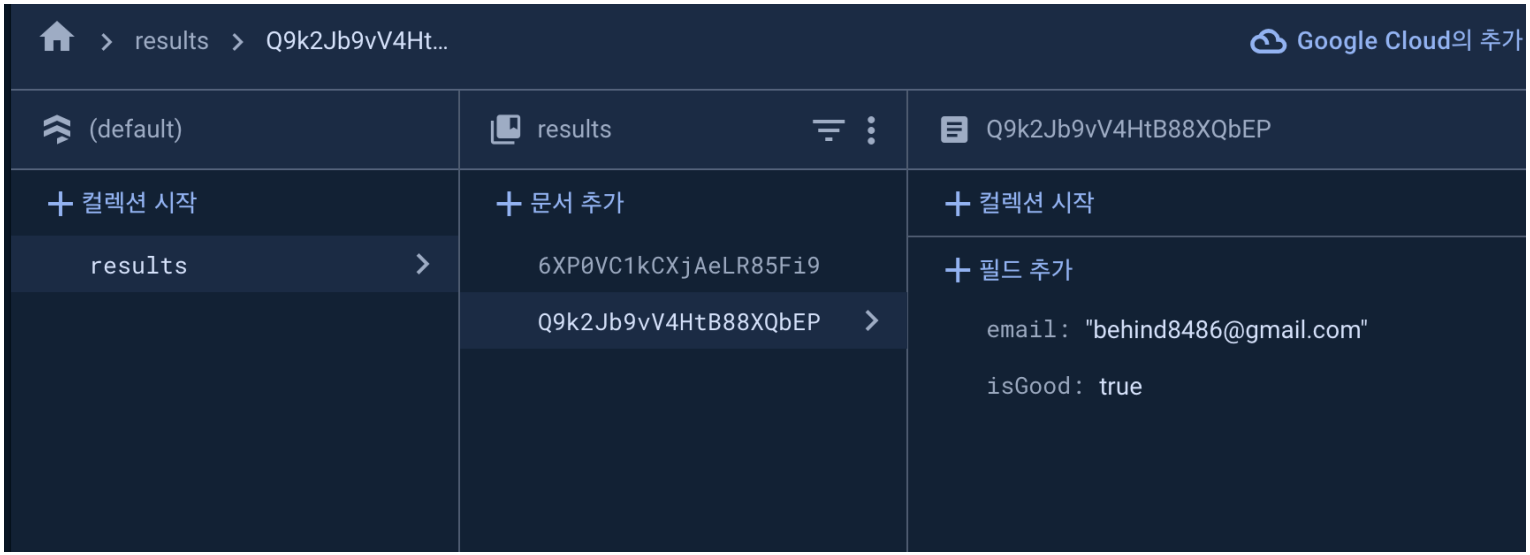
Application 구현

Process

Cloud Functions를 이용하여 Storage에
새로운 비디오 파일이 생성되면
AI 서버로 연산 처리 요청

4 Application 구현

Process



AI 서버로부터 연산 과정이 끝나면 Firestore 데이터베이스에 결과값을 저장
Cloud Functions에서 이벤트 트리거를 감지하여 새로운 값이
데이터베이스에 들어오면 해당 데이터를 조회

Application 구현

Process

[AI Speech Supporter] Your Result is Arrived now!

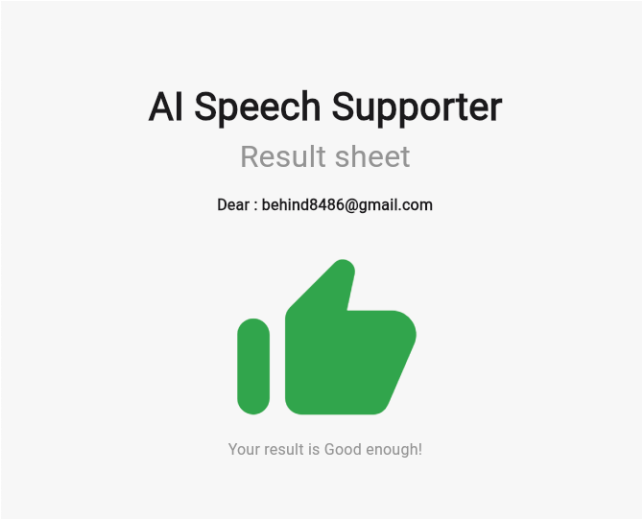
김동한 <behind8486@gmail.com>
나에게

Dear : behind8486@gmail.com

Thank you for waiting!

If you want to see your result, go to below link :

http://bit.ly/3syPhE7



Email로 전송하여 결과를 출력받을 수 있음

5

한계점

5 한계점

Dataset 증강에서의 한계점

추가적으로 좋은 데이터셋과 다른 특징을 보일 경우에 나쁜 데이터 셋이라고 가정
좋은 데이터와 다를 경우에 나쁜 발표라고 생각하고 데이터셋을 추가 제작

5 한계점

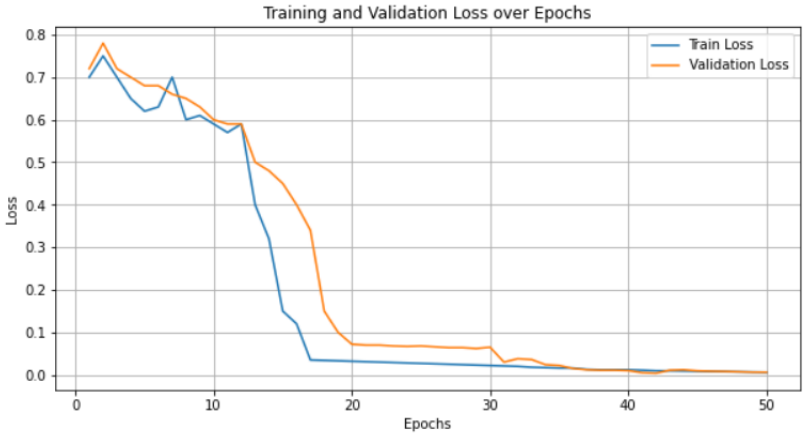
Dataset 증강에서의 한계점

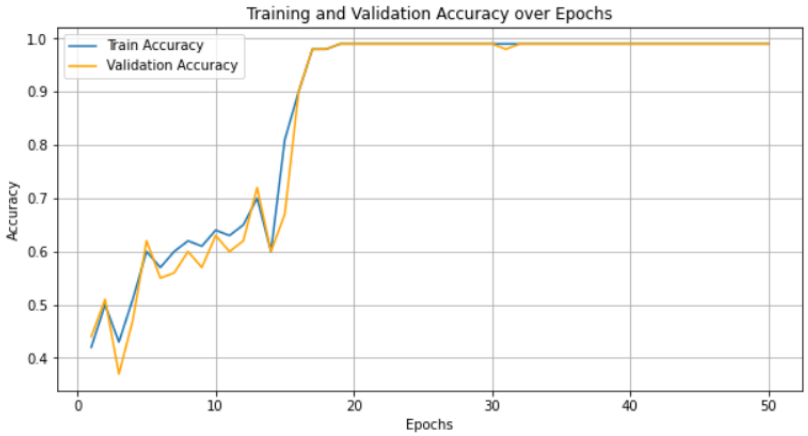

나쁜 데이터셋를 추가할 때, 특정 평가 요소를 가정하여 그 부분을 강조해서 제작하기도 함
특정 문제점만이 과하게 학습될 수 있는 문제점이 발생할 것이라고 예상

5 한계점

Dataset 증강 이후 모델의 한계점

데이터 증강 이후
#Data =1048



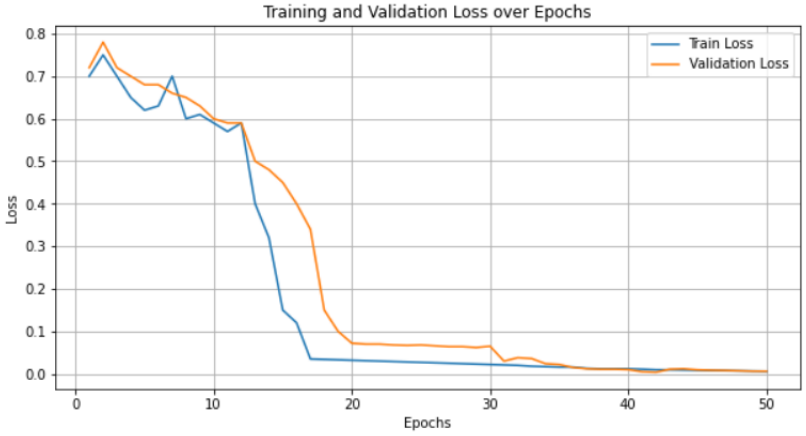


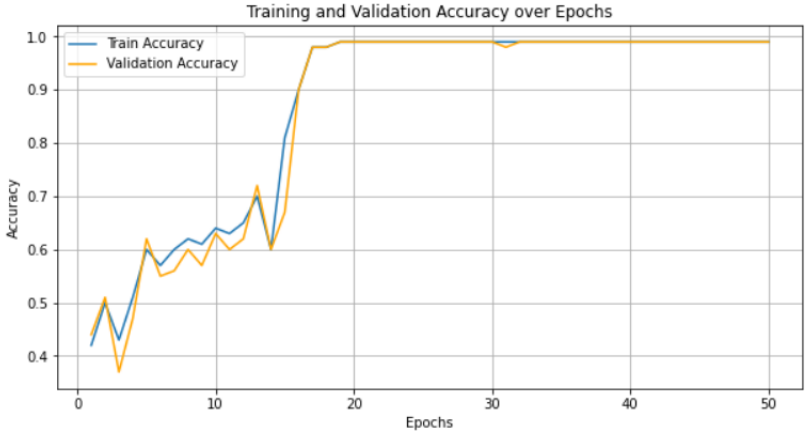
데이터 증강을 진행했으나, 여전지 작은 epoch에서 과적합되며, 진동이 발생
또한 Train 그래프를 너무 과하게 따라가는 Valid 그래프
Accurary를 너무 따라가는 모습이 보임, Scale 문제에 의해서 발생한다고 생각

5 한계점

Dataset 증강 이후 모델의 한계점

데이터 증강 이후
#Data =1048

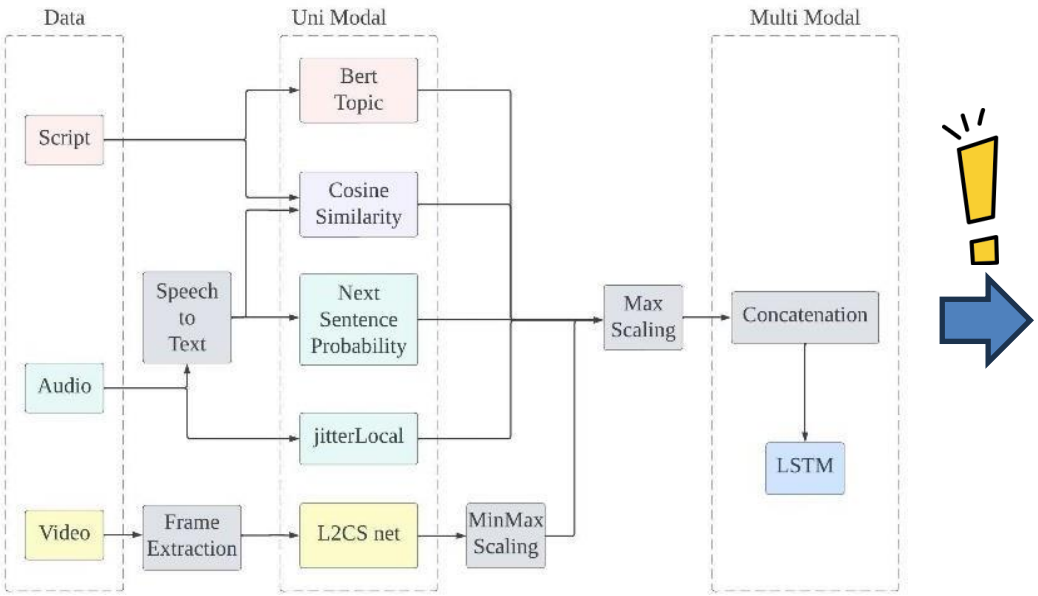


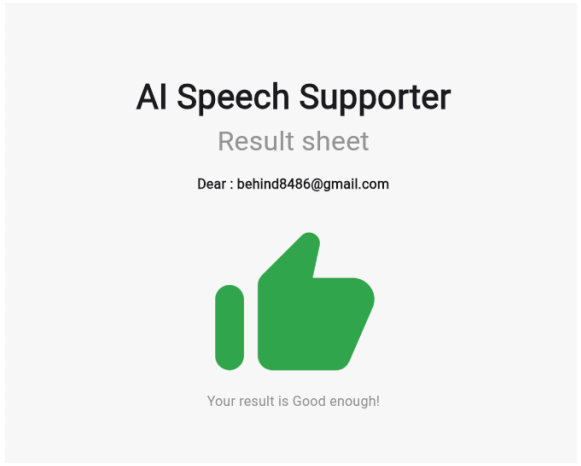


또한 더 적절한 데이터 셋을 추가하여 학습해야 할 필요성을 느낌
복잡한 모델을 적은 데이터로 학습하다 보니, 발생한 문제점이라 판단됨

5 한계점

해석력에 있어서의 문제점

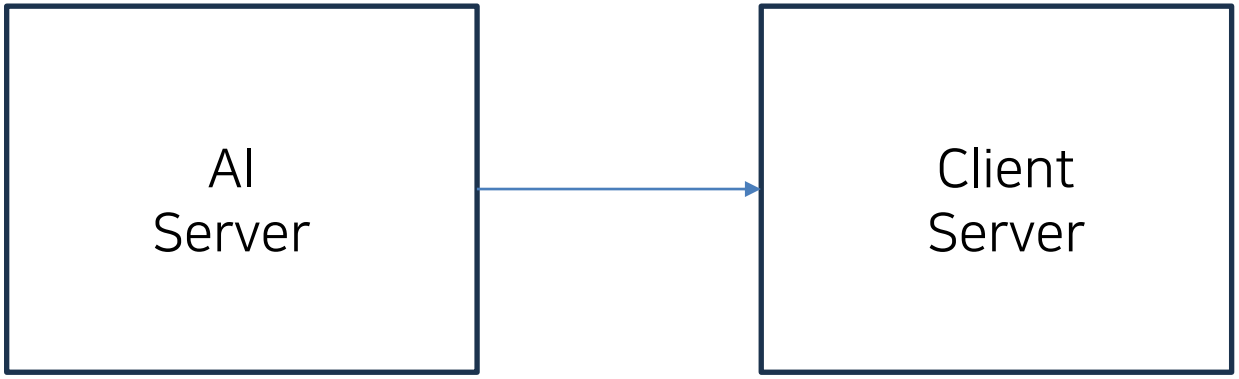




발표 평가 모델임에도 불구하고 왜 다음과 같은 평가가 나오게 되었는지 해석이 불가능함
다양한 해석적 모델을 통해서 모델링을 구현해보려 노력하였으나,
Stacked 될 경우에(혹은 앙상블) 각 해석이 용이하지 않음
추가적으로 각 모달에서 해석력 또한 얻기 쉽지 않았음

5 한계점

실생활 어플리케이션 배포에 있어서 문제점



실사용 하기에는 딥러닝 서버에서 결과까지 변환하는 과정에 있어서 오랜 시간이 걸림
변환 시간까지의 중간 시간을 어떻게 처리할 것인지 추가 고안 필요

THANK YOU!