

[DS 대본]

첫쪽

- (강사님 시작하라고 하면 그냥 바로) 카톡 대화 틀기
- (클릭) Six-sigma 팀 발표 시작하도록 하겠습니다
- (클릭) 안녕하세요 발표자 주경배라고 합니다. 저희 조는 표정 분석을 통한 감정 분류 및 음식/맛집 추천을 주제로 이번 프로젝트를 진행하였습니다.

목차

- 발표 순서는 프로젝트 개요, 프로세싱, 서비스 시연, 기대효과, 개발 후기 및 느낀점 순으로 진행하겠습니다

Part1 프로젝트 개요

- 첫번째 파트는 프로젝트 개요의 기획 배경 및 목표입니다

프로젝트 기획 배경 및 목표

- 저희 조는 딥러닝 프로젝트 기획에 있어 수많은 리서치를 해본 결과, 표정 인식 분야의 모델들이 거의 대부분 서양인 얼굴 데이터로 학습되어 서양인 얼굴에 특화되어 있다는 사실을 발견하였습니다.
- 이러한 문제점을 도전 과제로 삼아,한국인 얼굴에 특화된 표정인식 모델과 기존에 없었던 한국인 감정기반 추천 서비스를 만들기로 하였습니다.
- 또한, 이 도전적인 과제에 삶의 필수 요소인 “음식”을 저희가 만들 모델에 적용시켜 한국인 표정분석을 통한 감정분류 및 음식/맛집 추천으로 주제를 확장하기로 하였습니다.
- 다음은 음식메뉴추천 서비스라는 주제를 선정할 때 고려한 사항들입니다
- 첫번째로, 요즘 ‘점메추’와 같은 신조어가 생길 정도로 메뉴 선택에 대한 고민이 많아지고 있는 것을 파악해 저희 서비스의 충분한 니즈를 확인했습니다

- 두번째로, 유튜브, 넷플릭스와 같은 추천 시스템이 IT 업계의 트렌드로써 자리를 잡고 있기에 저희 서비스의 목표인 메뉴 추천 시스템의 방향성을 확고히 할 수 있었습니다
- 마지막으로 사업성입니다. 저희는 주제를 선정할 때 사업성과 대중성이 있으면 좋겠다고 생각을 했습니다. 그래서 요식업종에서 큰 입지를 가지고 있는 회사를 찾다가 백종원 대표가 운영하는 더본 코리아 브랜드로 메뉴를 추천하기로 정하였습니다.
- 그렇다면 누가 우리 서비스를 사용할 것이며, 메뉴를 선정할 때 추가적으로 고려해야 할 것이 무엇이 있을까 생각해 보았습니다
- 저희 서비스의 대상은 흔히 MZ 세대라 불리는 20~30/ 대학생, 취업생, 직장인으로 선정하였고
- 메뉴 추천에 있어 좀더 상황에 맞는 다양한 추천을 위해 감정 뿐만 아니라 날씨와 계절을 포함하게 되었습니다
- 저희 조는 Data Science 파트와 Data Engineering 파트로 역할을 분담해 프로젝트를 진행하였습니다
- 파트 구성원을 말씀드리면, DS 파트의 박은지님, 정찬영님, 주경배님, DE 파트의 박창훈님, 이원근님, 최인태님이 있습니다.
- DS 파트에서는 표정 분석 모델 생성 및/ 서비스 구현을 위한 모델 분석 임무를 맡았고
- DE 파트에서는 편의성을 고려한 화면설계와, 데이터 필터링 및 파이프라인 구축을 통한 서버구현을 맡았습니다.

Part2 프로세싱

- 두번째는 프로세싱 파트입니다

전체 흐름도

- 전체적인 프로세싱 흐름도는 데이터 수집에서 첫번째/ DS 파트의 모델링, 두번째/ 공통 파트의 메뉴 추천 알고리즘 구현, 세번째/ 두개의 산출물을 통한 DE 파트의 DB 구축 및 웹 구현입니다.

Part2 DS

- 이제 본격적으로 **DS** 파트로 넘어가겠습니다.
- 이 페이지는 **DS** 파트의 작업 흐름도로/ 간략하게 설명드리면, 데이터 수집 (클릭)/ EDA (클릭)/ 데이터 전처리 (클릭)/ 모델링 (클릭)/ 최종모델 선택 (클릭)/ 최종모델 분석 (클릭) 순으로 진행됩니다

Part2-1 데이터 수집

- 데이터 수집 파트입니다

출처

- 딥러닝 학습에 필요한 데이터를 수집하기 위해 **AIHub**를 이용했고, **836GB** 크기의 데이터 셋을 찾을 수 있었습니다
- 또한 감정 기반으로 음식을 추천하는 서비스의 근거를 이화여자대학교의 논문으로 하였음을 말씀드립니다.

데이터셋 생성

- 주어진 장비로는 원 데이터 처리가 불가능 했기에 각 감정당 이미지 데이터 **1,000**개를 목표로 타당성을 가지고 데이터를 줄여 나가는 작업을 진행했습니다.
- 먼저 논문과 **AIHub**의 감정중 중복되는 **4**가지 감정을 고려하고,가장 크기가 작은 **Validation set**을 선택하여, **52GB** 크기의 **4**가지 감정 **validation** 데이터를 확보하였습니다
- 위의 과정에서 줄여진 **3만3천여**개의 데이터를 각 감정별로 **1000**개씩 표본 추출하는 과정을 진행했습니다.
- 마지막으로, 이미지를 크롭핑해 최종 **3,520**개의 데이터 셋을 확보했습니다

이미지 크롭핑

- 저희는 모델 성능을 보다 높이기 위해 사람 얼굴만 자르는 **cropping** 작업을 진행했습니다. 보시는 바와 같이 원본 이미지는 배경과 다른 인물들이 나오기에 모델 학습에 있어 방해 요소가 될 것이라 생각해서 **autocrop**로 모든 이미지를 처리해 주었습니다
- 그 결과 **cropping** 실패 이미지를 제외하고 각 감정별 **880**장으로 축소할 수 있었습니다.

데이터 분석

- 다음은 데이터 분석 파트입니다

이미지 속성 분석

- 저희는 이미지 데이터셋의 **3**가지 속성 분석 해보았습니다.
- 첫번째로 나이는 분석결과, **2030**에 더 특화될 것이라는 것을 예측해 보았고
- 두번째로, 성별은 여자 이미지에 더 특화될 것이라 예측해 볼 수 있었습니다.
- 마지막으로 **RGB**는 감정별로 비율이 다르지 않아, 모델링에 유의미한 영향을 주지 않을 것이라 예측하였습니다

데이터 전처리

- 다음 전처리 파트는 **4**단계로 세분화 하였습니다

데이터 전처리 – train, validation, split

- 첫번째 스텝으로, 모델링의 학습과 검증을 용이하게 하기 위해 확보된 데이터셋을 **splitfolders**를 이용하여 **train, validation, test** 셋으로 각각 **6대2대2**로 분할하였습니다.

데이터 전처리 – imageDataGenerator

- 다음 **3**개의 스텝은 텐서플로우의 **imagedatagenerator**를 사용하여 진행하였습니다.
- 이미지를 **150 by 150**으로 **resizing** 해 주었고
- **1/255**로 정규화를 해 주었습니다
- 그리고 각 폴더마다 **one-hot-encoding**으로 레이블링을 해 주었습니다

Part 2-4

- 이제 모델링 파트로 넘어가겠습니다

목표치 설정

- 모델링을 하기에 앞서, 만들 모델의 정확도 목표를 먼저 설정하였습니다.

- 목표치를 설정하는데 있어 **Kaggle**의 표정 분석 대회와, 표정분석과 관련된 논문들의 정확도를 참조하여 그 중간 값인 (클릭) 80% 이상의 **Accuracy**로 목표를 설정하였습니다

모델링 – baseline

- 모델링의 처음 단계로는 **baseline** 모델을 만들어 보았습니다
- 위 그래프는 우측부터, 기본 모델, **cropped**가 추가된 모델, 이미지 사이즈 **250by 250**로 변환된 모델, 마지막으로 **augmentation**이 추가된 순서의 모델 **accuracy**를 보여주고 있습니다.
- 그래프에서 나타나지듯 마지막 모델이 가장 좋은 성능을 보여주었고 이러한 모델의 이미지 설정 값을 뒤에 나올 모델에도 동일 적용하였습니다.

모델링 – 전이학습

- 모델의 성능을 보다 높이기 위하여 전이학습을 진행하였습니다.
- 이미지 인식 경진대회에서 우수한 성능을 보여준 다양한 모델들을 이용하였습니다

모델링 – fine tuning

- 다음은 전이학습 모델의 미세 조정에 대한 그래프입니다
- 전반적인 전이학습 모델이/ 동일한 양상으로 **hyperparameter** 값을 조정할 때 성능이 높아지는 것을 보여 주었고, 이 페이지에서는 대표적으로 **VGG16** 모델로 설명 드리겠습니다
- 맨 좌측 **basic** 모델의 (클릭) **gradient exploding** 문제로 인해 **accuracy**가 25% 밖에 나오지 않은 것을
- 두번째에서 **Learning Rate**을 줄여 줌으로써 75% 까지 올릴 수 있었고,
- 세번째 부터는 (클릭) **overfitting** 문제를 잡기 위해 **dropout**, **regularization**, **batch normalization**을 사용하였습니다
- 그중에서 **dropout**과 **batch normalization**을 같이 사용했을 때 모델 성능을 가장 극대화 할 수 있었습니다 (클릭)

Weights = None

- 전이학습을 위해 추가적으로 고려한 사항은 **Weight** 하이퍼파라미터 입니다. **Weights**에 주로 사용하는 **imagenet** 값은 **1000**개의 사물과 동물의 분류를 합니다만, 저희 모델은 ‘사람의 표정’에 초점을 맞춘 모델이기에 **none** 값을 넣어 시험해 보았습니다.
- 보시는 바와 같이 **Accuracy** 또한 **80%** 웃도는 모습을 보였지만 ,**epoch**가 **500** 번을 넘어가며 학습 시간이 **5시간** 이상이 걸리는 양상을 보여주어
- **weight**를 **none**으로 두는 방법은 효율적이지 않기 때문에 (**클릭**) 기존에 적용하였던 **imagenet**이 적합하다는 것을 확인하였습니다.

Part 2-5 최종 모델 선택

- 최종 모델 선택입니다
- 저희는 모델을 선택할때 높은 성능만을 보는 “웹” 환경과, 모델의 용량도 같이 보는 모바일 환경을 각각 고려하였고
- 첫번째로 **Parameter** 값만 고려했을때는 **SE ResNet18** 모델이 앱 환경에서 가장 적합하다고 선정되었고
- 두번째, **Accuracy** 만을 고려했을때는 값이 가장 높은 **VGG16** 모델이 웹 환경에서 가장 적합하다고 선정되어 저희 서비스에 적용하게 되었습니다

Part 2-6 최종 모델 분석

- 이제 최종 모델 분석 파트로 넘어가겠습니다

Fine Tuned VGG16

- 지금 보시는 화면 최종 모델의 세세한 정보를 담고 있습니다. 우측은 학습을 진행함에 따라 **Loss** 와 **Accuracy**가 변화하는 그래프를 보여주고,
- 좌측은 **confusion matrix**로 각 감정의 실제 예측 값을 보여주고 있습니다

Grad-Cam

- 저희 **DS** 파트가 모델 분석에 있어 가장 심도있게 본 분야가 바로 **Grad-Cam** 입니다. **Grad-Cam** 분석은 이미지의 어떤 부분이 모델에서 가장 집중적으로 고려되어 지는 가를 히트맵을 통해 보여주는 기법으로 모델의 심층적인 분석을 가능하게 합니다

- 보시는 페이지는 실제로 저희 모델에서 이미지의 특징을 추출하는 12개의 **convolution layer**를 해체하여 히트맵으로 시각화 한 것입니다.
- 어떠한 식의 변화가 있는지 같이 한번 보시겠습니다
- 동영상 재생 - 첫번째 **layer**에서 시작하여,,,,,,,,, 마지막 **layer**까지 변화하는 모습입니다... 후반으로 갈수록 얼굴표정의 정확한 부분을 모델이 캐치하고 있습니다.
- 저희는 마지막 **layer**를 각 감정별로 시각화 하였고 그 결과 의미있는 **outcome**을 도출 할 수 있었습니다
- 저희 모델은 얼굴의 근육을 많이 쓰는 “분노”와 “기쁨”에서는 얼굴 중앙부와 입가를 잡아내는 양상을 보여주었고
- 표정이 비교적 과하지 않은 “슬픔”과 “중립”은 주로 입 이라는 한정적 부분을 보고 감정을 판단하는 모습을 보여주었습니다.

Model Test - EDA 가설 검증

- 다음으로 저희가 한것은 **Model Test**로 앞서 만든 **EDA**에서의 가설을 검증하였습니다 .
- 그 결과 저희가 세운 가설과는 다르게 남겨 이미지의 별다른 차이를 보이지 않는다는 사실을 알아냈습니다
- 하지만 나이에서는 저희의 가설과 동일하게 모델이 **2030** 이미지를 **4050** 보다 더 우수하게 잡아낸다는 것을 알 수 있습니다

Model Test - 가설 검증

- 또한 **cropped**된 이미지와 그냥 사진들의 성능 차이를 비교해본 결과 **cropped**된 이미지가 가로나 세로가 긴 이미지에 비해 정확도가 높은 것을 보실 수 있습니다. (클릭)

모델 시연 - opencv

- 마지막으로 저희는 서비스 구현에 앞서 모델이 잘 돌아가는지 보기 위해 **opencv**에서 실제 이미지를 가지고 구현해보았습니다.
- 이 영상은 컴퓨터 웹캠에 보여지는 태블릿 이미지에 대해

- 저희 모델의 구현이 어떠한 식으로 이루어지는지 보여주는 시연 영상입니다. (기쁨, 분노, 중립, 슬픔을 차례대로 보여주고 있습니다)
- 이상으로 **DS**파트 발표를 마치겠습니다

메뉴 추천 알고리즘

- 다음은 메뉴 추천 알고리즘 파트로,
- 저희 팀에서 공통으로 실행한 메뉴 추천 알고리즘 부분에 대해 말씀드리겠습니다

감정 & 날씨에 따른 카테고리별 메뉴

- 저희는 감정 및 날씨에 따른 카테고리별 메뉴를 보이시는 이 두 논문을 통해 가져왔습니다.
- 각 논문마다 상황별 음식 카테고리가 1~10순위 까지 나와있음을 보실 수 있습니다.

Menu Categorization

- **DE** 파트에서 수집해 주신 더본 메뉴 데이터를 가져와 (클릭) 모든 **DS/DE** 구성원들이 힘을 합쳐 논문의 카테고리별 라벨링 작업을 해 주었습니다.

메뉴별 가중치 계산

- 마지막으로 위에서 라벨링 작업을 해준 각 메뉴들을 3가지 단계로 가중치를 계산해 주었습니다.
- 첫번째, 모든 경우의 수를 계산해 27가지의 감정, 계절, 날씨의 경우의 수를 산출해 냈고
- 두번째, 논문에 따른 상황별 순위로 각각 점수를 매겼습니다
- 마지막 세번째, 두번째에서 점수 매긴 상황들을 모두 계산하여 가중치를 주었습니다.
- 이렇게 가중치로 인한 순위가 매겨진 메뉴들은 **DB**에 탑재되었고 음식을 추천할때 1순위 부터 5순위 까지 바로 바로 나오는 서비스의 일환이 되었습니다
- 다음 **DE** 파트는 최인태님께서 이어서 발표해 주시겠습니다

[DE대본]

안녕하세요 DE Part 발표를 진행할 최인태 입니다.

저희는 DS파트와 공통 파트의 결과물을 어떻게 구현할 수 있을까 생각하다, 장고를 사용하여 웹서비스 형태가 가장 적합하다 결론을 내렸습니다

[워크플로우 44p 화면비추기]

저희의 의사결정 및 실행단계는 총 4단계로 이루어져있으며,

각 해당 내용에 대한 간단한 언급만 말씀드리고, 뒤에서 자세히 말씀드리겠습니다.

데이터수집단계에서는 정보수집 , 데이터 전처리 단계에서는 데이터 분류 및 정제작업, DB설계 및 구축 단계에서는 DB모델링을 시행하였으며, LOCAL SERVER 구축 단계에서는 웹서비스를 위한 장고 프레임워크를 사용하여 서버를 구축 및 화면설계 실행하였습니다.

[44p 끝]

[45p 비추기]

저희가 장고를 사용하기 전에 제일 고민했던 부분이 두가지가 있었는데,

이미지처리와 EMOTION MODEL(표정분석알고리즘)을 어떻게 장고에 탑재할 것인가에대한 고민이었습니다.

첫번째 고민인 이미지처리에 대한 해결방법을 탐색중에 장고내부의 정적폴더와 DB를 사용하여 저장하는 방식으로 해결하였으며, 이 방식은 장고는 STATIC이라는 정적폴더에 이미지파일을 저장하고, DB에는 IMGURL(이미지주소, 즉 파일명)을 저장하는 방식을 택합니다.

이를 적용한 전체적인 데이터흐름은 다음 화면과 같습니다. 클라이언트가 그림파일을 입력하면, (클릭)장고내부에서는 (클릭)이모션모델(표정분석 알고리즘)과 함께 날씨크롤러가 작동하여 가중치를 부여한 감정분석 결과와 음식추천 메뉴를 웹페이지형태인 HTML에 전송하고 화면에 데이터를 표시합니다.

이와 동시에 DB에서는(클릭) IMG URL(파일명)을 저장하며, 음식추천 및 음식메뉴 가중치 테이블에 데이터를 저장하거나, 데이터를 추출해 (클릭)다시 장고(서버)로 돌려주며, 장고는 이 데이터를 CLIENT 의 화면에 웹페이지 형태로 보여줍니다.

[45p 끝]

[46p 비추기]

표정분석 알고리즘을 사용하기 위해서는 이미지 전처리(리사이징)이 필수적으로 필요했습니다.

이를 간단히 도식화 하면 다음화면과 같습니다. 이미지를 입력하고 (클릭)리사이징을 하며 정적폴더 및 지정된 폴더에 이미지 파일, 그리고 DB에 주소를 담아필요할때마다 이미지를 꺼내 쓸 수 있도록 지정된 형식입니다.

필로우라는 라이브러리를 사용하여 이미지 크기를 조절할 수 가 있었고,크기가 조절된, 전처리가 완료된 이미지를 해당 알고리즘에 투입할 수 있어, 좋은 결과를 도출할 수 있었습니다.

두번째 문제인 모델 탑재는 장고 내부의 함수인 load_model을 사용하여 생각보다 간단히 처리할 수 있었습니다.-[46p 끝]

[47p 시작]

이제 본론으로 들어가서 첫번째 단계인 데이터 수집단계에서는 저희는 장고를 사용하기 전에 저희는 정보수집, 즉 데이터 수집이 필요하다고 느꼈고, 저희가 배운 내용을 최대한 활용하기로 결정함에 따라, 크롤러를 제작하기로 결정했습니다.

[47p 끝]

[48p 시작]

저희는 더본코리아를 타겟으로 설정함에 따라, 각 브랜드별 메뉴정보를 수집할 수 있는 크롤러를 재사용이 가능한 함수형태로 제작함으로써 추후에 재사용이 가능하도록 설계했습니다.

따라서 저희는 크롤러를 사용하여 더본의 브랜드별 매장의 메뉴정보 크롤러, **[49p 시작]** 그리고 인간의 표정이나 감정이 날씨에 영향이 있다는 논문에 따라 날씨정보 크롤러를 만들었으나, 이 크롤러는 별도의 데이터수집이 없이, 장고안에 심어 서버가 실행되어있으면 언제든지 당일날의 날씨를 크롤링해 추후에 나올 메뉴정보추천시에 포함되도록 알고리즘을 구축했습니다.

[50p 시작]마지막으로는 지도위에 매장 정보를 표시하기 위한 매장 정보크롤러 등의 수집을 위하여 크롤러를 제작하여 브랜드명, 지역, 지점명, 도로명주소, 전화번호, 영업시간, 해당 브랜드 매장의 **[51p시작]**위도 경도 정보를 수집하여 지도위에 표시할 수 있는 정보 등을 수집하였고, **[52p시작]**총 16개의 매장에서 348개의 메뉴를 수집할 수 있었습니다. 상단의 크롤러는 모두 별도의 파이썬 프로그램으로 만들어 외부에서 동작할 수 있게 설계했습니다.

[53p]두번째 단계에서는 원하는 데이터를 추출하기 위해 전처리 과정을 거쳤습니다.**[54p]** 특히 메뉴정보는 각 매장별로 정리 및 중복메뉴를 제거하는 과정이 필요했고, 앞서 DS파트에서 언급드렸던 메뉴를 라벨링 과정을 거쳐 DB에 입력(임포트)시킬 수 있었습니다.

이에 따라 데이터 전처리가 완료되어 다음 단계로 진행을 할 수 있었습니다.

[55p] 세번째 단계인 DB 구축 및 모델링 단계에서는

[56p]이후 저희는 DB구축을 위한 DB모델링 및 ERD를 제작했으며, 해당 모델링대로

장고에 DB model을 구축하였으며, DB안의 데이터를 가져오기위한 쿼리문을 제작했습니다. DB테이블은 총 4개로 이루어져있으며, 이미지를 처리할 수 있는 테이블, 총 추천결과를 저장할 수 있는 테이블, 그리고 가중치와 메뉴정보가 담겨 있는 테이블, 그리고 메뉴선택에 따른 추천결과 저장 테이블로 이루어져있습니다.

이로써 서버구축 및 화면구현을 위한 사전작업(데이터 수집 및 정제)과정이 모두 끝났기에,

[57P 시작]마지막 네번째 단계인 로컬 서버 구축 단계및 화면구현 단계에서는 어떻게 하면 효율적으로 서비스를 제공 및 구현 할 수 있을까에 대한 고민을 시작하였고, 웹서비스의 설계도인 흐름도부터 제작하기로 했습니다.

[58P]지금 보시는 화면이 저희가 설계한 웹서비스 흐름도 입니다.

처음에 메인페이지에 접속을 하면, 이미지 수집동의 화면을 팝업으로 띄우는 데, 이는 저희는 효율적이고 명료한 서비스를 구현하기 위해 회원가입을 별도로 받지

않는대신, 저작권 및 초상권 문제를 대비한 간단한 동의과정을 거치도록 설계했습니다.

만약 동의를 누르면(클릭) 이미지를 업로드 할 수 있고, 비동의를 누르면 메인화면(첫서비스)화면으로 리다이렉션 됩니다.

표정분석이 된다면 (클릭)바로 결과추출 페이지로, 표정을 분석할 수 없다면 표정을 선택할 수 있는 페이지(molar.html)로 접속해 기쁨/슬픔/행복 중에 하나를 선택해서 데이터를 전송할 수 있게 제작했습니다. 그렇게 추천된 결과는(클릭) show_food.html이라는 곳에 데이터가 전달되어 db축적 및 화면에 출력합니다.

서버구축 단계에서는 저희는 이미 전처리된 데이터를 DB에 입력 해놨기 때문에, DB있는 데이터를 최대한 활용하기 위해 각종 함수 설계 및 코딩을 시도했으며 여러번의 시도 및 실패 끝에 데이터를 DB로부터 끌어와 화면에 출력하거나, 클라이언트(유저)의 데이터를 입력받아 실시간 전처리 및 DB에 입력하는 방식을 찾아 적용했습니다.

[59P시작]그결과 UI는 최대한 사용자의 편의성 및 편리성에 따라 심플하게 설계하도록 노력했고, 한눈에 모든 메뉴가 들어오도록 설계했습니다. 강조할 부분에 포인트(폰트색깔 크기등)를 두어 최대한 간편하면서도 쉽게 알아 볼 수 있게 제작했습니다.

그 결과 지금 화면에서 보시는 것처럼 사용자가 1~5순위까지 추천메뉴중 하나의 메뉴를 선택하게 되면 저희는 사용자의 감정,날씨,계절, 추천음식, 음식점 등의 메뉴를 자동으로 입력받을 수 있게 DB를 및 프로그램을 설계했습니다

[60P]

[61P]그럼 약 2분간의 웹서비스 시연 영상 재생 후에 다시 뵙도록 하겠습니다.

[62P] 영상 잘 보셨나요?

[63P] 저희 서비스의 예상되는 기대효과는 클라이언트(고객)은 음식 선택에 대한 고민이 줄어들 것이고, 감정을 배려한 메뉴에 고객만족도가 상승할 것이고, 자연스레 해당 매장의 매출이 증가로 이어질 것입니다.

[64P] 저희 팀이 설정한 -앞으로 서비스의 발전 방향으로는 ① 메뉴판 QR제작 ② 키오스크 웹캠 스트리밍 ③ 앱서비스 제작 ④ NLP기반 감정분류 ⑤ 로그기록 기반으로 추천 알고리즘 강화 등의 의견이 있었고,

개선사항 및 문제점으로는 DS파트에서는 ① 장비역량(저장공간,GPU)의 저하문제 ② 네가 감정이외의 미세한 표정차이 추가학습이 필요 ③ 원본이미지도 가능하도록 face detection model 추가(얼굴부분 확대 및 축소 없이 가능하게) 등이 있었으며

DE파트에서는 ① 서비스만족도 기능 추가 ② LOCAL서비스가 아닌 외부배포기능추기 ③ 추후 로그기록에 따라 DB보완등의 개선사항 및 문제점 등이 있었습니다.

[65P] 다음 화면은 수 행 도구 들입니다..

[66P] 또한 저희는 이 프로젝트를 진행하면서 사진과 같이 학원에서 대면 모임을 가졌었고, 프로젝트의 주제, 진행방향, 일정, 그리고 디자인 등의 방향을 논의했습니다. 다수결에 따라 결정하는 것이 아닌 개개인의 의견을 존중하고 반영했습니다. 그리고 서비스가 어느정도 완성된 뒤에는 전체적인 테스트를 겸한 표정분석 모델 테스트도 할겸 저희가 만든 서비스를 이용해 음식을 추천받아 점심메뉴를 선정하였고, 식사시간을 함께 가졌습니다.

또한 온라인에서는 매일 5시30분의 의견을 나누는 등, 팀원들과 공유하는 시간을 많이 가졌고, 많은 의견을 나눴기 때문에 충돌없이 원만히 마무리 되었습니다. 모든 팀원분들께 감사의 말씀을 전합니다.

[67P] 마지막으로 보시는 화면은 저희가 개발을 진행하면서 느낀점입니다.

이상으로 발표를 마치겠습니다. 경청해주셔서 감사합니다. 좋은하루 되세요.

[68P] 다음으로 QNA 시간을 가지도록 하겠습니다.

[녹화 대본]

3. 시연영상에 담을 내용 논의(어떤 페이지 보여줄 것인지, 어떤 사진 넣을 것인지)
 - 홈페이지
 - 동의 비동의 화면.
 - 비동의 누르면 다시 홈으로

- 동의 누르기
- 이미지 업로드 화면.
 - 중립 사진 한장 넣기
- 중립 시 감정 선택하는 화면
 - 기쁨 선택하기
- 메뉴 추천 화면
 - 추천된 다섯 개 메뉴 중 하나 선택 하는거 보여주기
 - 선택하면 매장 지도 화면으로 이동(로그데이터도 쌓인다고 어필하는 멘트 한번 해주기)
- 네비바에서 다시 추천받기 탭 누르기
 - 감정분류 명확히 되는 사진 한장 넣기
- 메뉴 추천 화면
 - 추천된 다섯개 메뉴 중 하나 선택하는거 보여주기
 - 선택하면 매장 지도 화면으로 이동(로그데이터도 쌓인다고 어필하는 멘트 한번 해주기)
- 네비바에서 “추천기록보기” 탭 눌러서 로그데이터 담기는 페이지 한번 보여주기
- 네비바에서 “매장정보보기” 탭 눌러서 로그데이터 담기는 페이지 한번 보여주기
- 최종 편집 러닝타임 : 2분 안쪽으로
- 끝