1. 기술 개발 전략
2. 데이터(수집, 전처리, 정제 방법)
3. 데이터 수집

많은 음식 사진 데이터를 직접 모으는 것은 불가능하므로, food 101 project의 파일 사용 예정

Food 101 data란?

총 101000개의 이미지와 101개의 카테고리가 있는 음식 데이터셋

데이터 다운 링크

<https://data.vision.ee.ethz.ch/cvl/datasets_extra/food-101/>

1. 데이터 전처리

food 101 dataset 사용할 경우 전처리가 되어있어 따로 할 필요 없음.

요약- Food 101 데이터 세트 다운로드 및 추출 -> 전처리랑 정제 따로 필요 없음

1. 구현 계획
2. 음식 인식-CNN 사용

Keras 및 기타 딥 러닝 라이브러리는 사전 훈련된 모델을 제공한다. 이는 ImageNet과 같은 데이터 세트에 대해 이미 교육된 효율적인 아키텍처(예: VGG, Inception, ResNet)를 가진 심층 신경망

사전 훈련된 Finetune Inception v3 모델을 조정하여 새로운 데이터셋에 retraining

\*Fine tuning이란?

Fine Tuning: 기존에 학습되어져 있는 모델을 기반으로 아키텍쳐를 새로운 목적(나의 이미지 데이터에 맞게)변형하고 이미 학습된 모델 Weights로부터 학습을 업데이트하는 방법

음식 분류기(어떤 음식인지)로 나눔

-> 우리는 음식 이미지만 사용할 계획이므로, 음식 검출기(음식인지 아닌지)는 필요X

Convolution Layer에서 추출된 사진 특징을 Neural Network에서 분류 결과를 산출 해냄(예측해낸 결과 몇가지와 각 항목과 일치할 확률) 그 중 가장 확률이 높은 것을 채택(레이블링 스무딩)

Data Augmentation을 통해 부족한 데이터를 보완 및 학습 강화(인식 성능 개선)

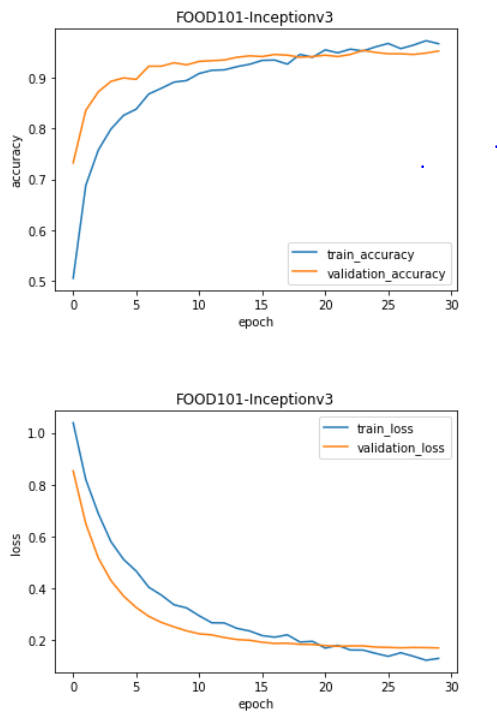
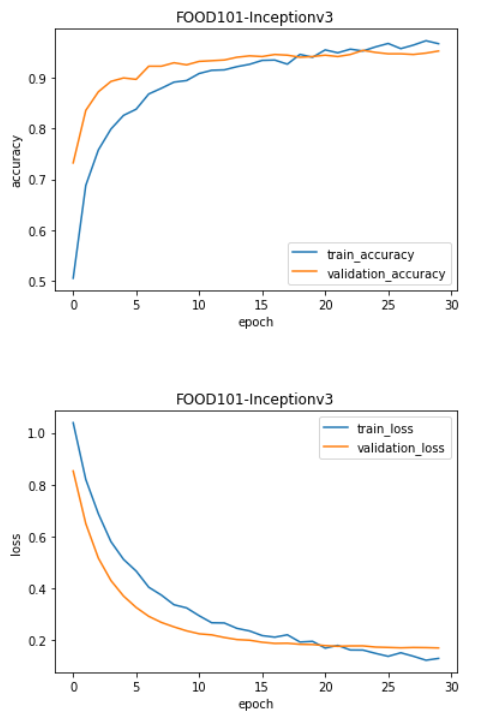
\*Data Augmentation이란?

Data Augmentation: 학습 이미지의 수가 적어서 overfitting이 발생할 가능성을 줄이기 위해 기존 훈련 데이터로부터 미세하게 변화를 주어 랜덤하게 생성하여 데이터의 수를 늘리는 방법

(학습 데이터 양을 늘리기 위해 기존 이미지에 변형(좌우 뒤집기, 자르기 등)을 가하는 방법)

Top-1 accuracy를 통해 성능 확인 / 정확도 및 손실 측정

정확도는 높고 손실은 적게



\*Top-1 accuracy란? CNN 성능의 평가 지표이다.(정확도)

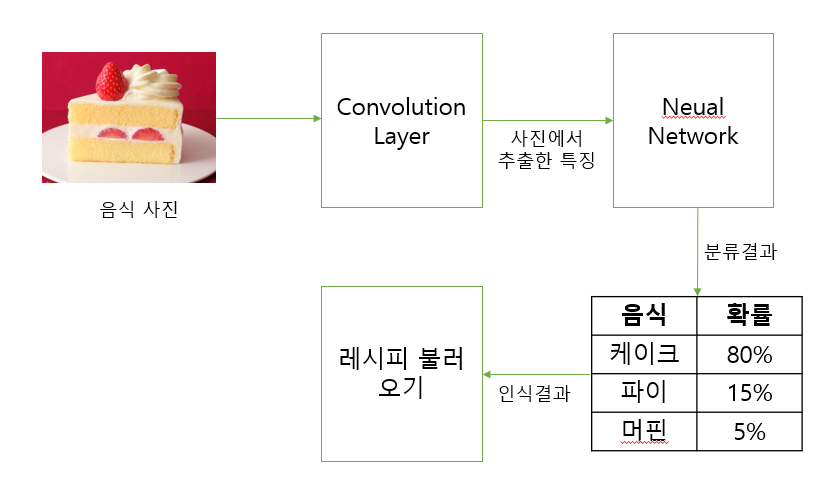
테스트 단계 -> 새로운 데이터를 사용하여 테스트

1. 레시피 불러오기

음식 분류기에서 판단한 음식명에 대한 레시피를 레시피 DB에서 불러옴

데이터를 수집하여 레시피를 담고 있는 데이터베이스를 만든 다음 이 데이터를 사용해 패턴을 찾고 음식 이미지와 해당 음식에 들어가는 재료 및 요리법을 연결하는 신경망을 학습

1. 예상 구조



1. 기술 조사
2. 음식 인식-CNN 사용(*InceptionV3* -인공신경망 GoogLeNet)

CNN이란?

1)Convolution layer를 통한 합성곱 연산으로 특징(feature) 추출하여 feature map 생성하고, 2)Pooling layer를 통한 feature map의 차원 축소(down sampling) 후 3)fully connected layer를 통한 이미지 최종 분류하는 방법

=> convolution layer와 pooling layer은 이미지로부터 한 단계 더 높은 추상화된 특징을 추출한 후 그 추상화된 정보에서 가장 중요한 정보만 남기도록 1/4크기로 압축하는 과정이 수십 번 반복되는 구조이다. 그리고 fully connected layer에서 추상화되고 요약된 정보로부터 어떤 물체인지를 인식, 분류하는 것이다.

\*CNN의 종류

LeNet, AlexNet, VGG Nets, GoogLeNet, ResNet 등이 있음

1. 레시피 불러오기-레시피DB에서 음식명에 해당하는 데이터를 불러옴
2. 기업 조사 -> 제공하는 서비스 중점으로 설명

카카오엔터프라이즈 AI Lab 음식 인식 기술

카카오 VX가 만든 ‘스마트홈트’ 또한 사진 속 음식의 이름과 칼로리를 자동으로 입력해주는 식단카메라 기능을 제공하고 있음. 이 식사 기록을 토대로 스마트홈트의 AI코치가 적절한 운동과 식단을 제안해줌.

음식 인식 기술: 검출기(이미지에서 음식 영역 추출)와 분류기(음식 이름 인식)로 나눔

-음식 분류기

1. 데이터셋 수집

스마트홈트 사용자가 많이 입력할 법한 음식명 목록화 후 음식별 후보 이미지 수백장을 가지고 정제 작업 진행

1. 동일한 음식 지칭하는 레이블 -> 하나로 통일

2. 같은 음식 but 다른 플레이팅 -> 같은 레이블 부여

3. 서로 다른 음식이지만 외관 비슷 -> 대표 범주 외 나머지 제거

텍스트이(가) 표시된 사진

자동 생성된 설명

1. 모델 채택 및 훈련

InceptionV4를 음식 분류를 위한 기반 모델로 활용

1. 최적화 알고리즘(Optimizer)

초반 ADAM 사용, 후반 Adabound 사용

2. 레이블링 스무딩- 일정한 값을 더하거나 빼는 방식으로 여러 범주의 인덱스에 값을 표시(multi-label distribution)

N개의 범주와 완전히 연결된 Fully connected layer를 통과시켜 점수(확률)가 가장 높은 범주를 선택

<작동 방식>

레이블이 잘못 부여된 데이터를 검수하는데 사용한 이미지를 통과시켰을 때, 여러 개의 예측 음식이 뜨는데, 그 중 가장 확률이 높은 것을 채택

3. 데이터 어그먼테이션

학습 데이터 양을 늘리기 위해 기존 이미지를 좌우로 뒤집거나 잘라서 사용하여 범주별 데이터 수를 균일하게 만듦, 이미지의 명도만 무작위로 조절(color jittering)하는 기법을 적용

4. 학습률 조절

에폭(epoch)마다 손실(loss) 값이 줄지 않으면 학습률을 1/2씩 줄임

5. TTA(Test-Time-Augmentation, 테스트 단계에서의 어그먼테이션)

서로 다른 어그먼테이션 기법을 적용한 테스트 데이터를 최종 모델(훈련 데이터와 검증 데이터로 학습을 마친 모델)에 입력

1. 제공하는 서비스

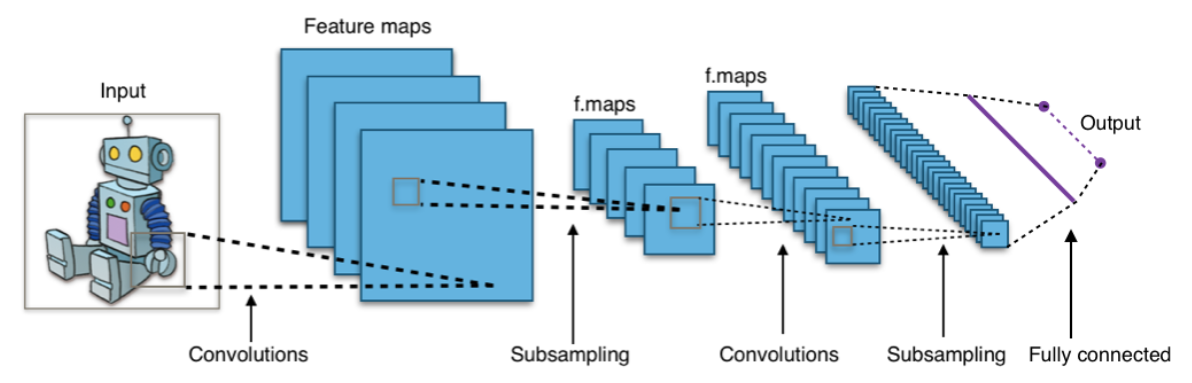
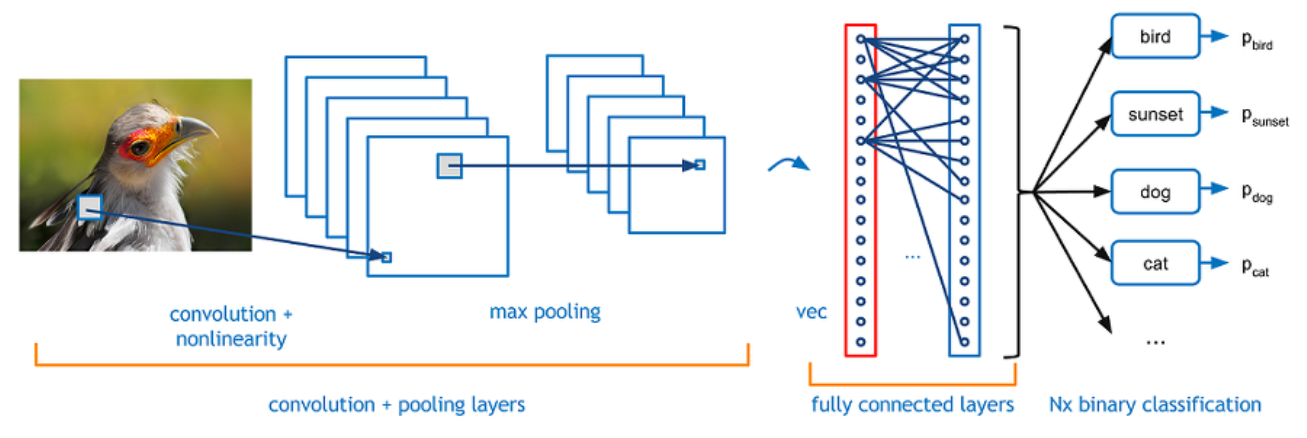
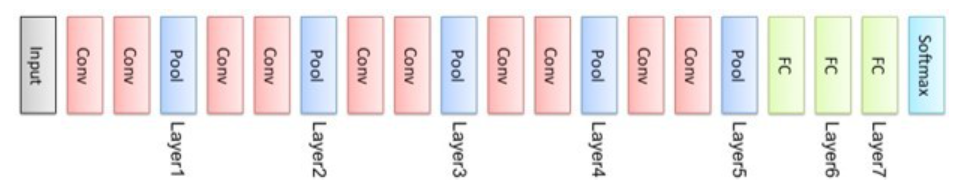
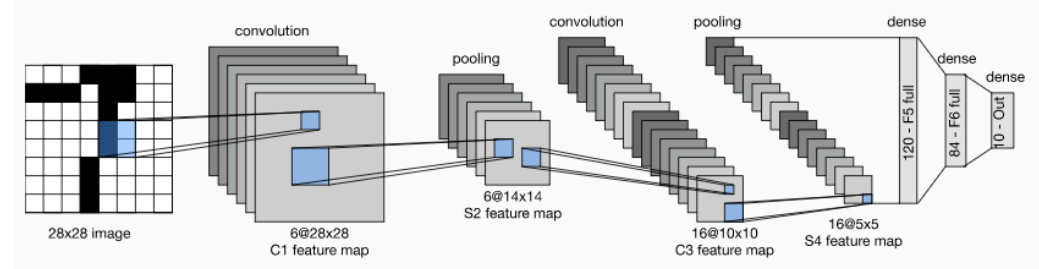
카카오 VX가 만든 ‘스마트홈트’ 또한 사진 속 음식의 이름과 칼로리를 자동으로 입력해주는 식단카메라 기능을 제공하고 있음. 이 식사 기록을 토대로 스마트홈트의 AI코치가 적절한 운동과 식단을 제안해줌.

1. 사업화 전략

요리 사업을 하는 업체에서 자신의 레시피를 실어달라고 의뢰받음 -> 회사 측에서 검토 후 레시피를 제공하기에 적절하면 사용자로부터 해당 음식의 이미지를 제공받으면 그 레시피가 뜸 (광고비를 받을 수 있음. 사용자는 입증된 레시피를 얻을 수 있고, 사업자는 자신의 사업(블로그, 유튜브 채널 등)의 홍보 효과를 기대할 수 있음

건강한 레시피/일반 레시피를 2가지로 나누어서 음식 검색할 때, 다이어트 버튼 선택 시, 그에 맞는 식단을 알려줌(카카오를 보고 이러한 사업화 전략을 생각해냄.)

1. CNN 시각자료

CNN원리 구조에 대한 그림 4개 중에 하나 사용

– Convolution layer를 통한 합성곱 연산으로 특징(feature) 추출하여 feature map 생성

텍스트, 고양이, 포유류이(가) 표시된 사진

자동 생성된 설명

– Pooling layer를 통한 feature map의 차원 축소(down sampling)

테이블이(가) 표시된 사진

자동 생성된 설명텍스트, 시계이(가) 표시된 사진

자동 생성된 설명

– fully connected layer를 통한 이미지 최종 분류

1. CNN 시각자료

**\*train\_model에서 중심적으로 봐야할 부분**

Tesorflow 안에 keras를 사용

Inceptionv3 모델 사용.

ImageDataGenerator()를 사용하여 train data를 augmentation 해주어 데이터 양을 늘린다.

flow\_from\_directory()를 사용하여 train과 test dataset을 generator 로 만들어 준다

pretrained 된 model인 inceptionv3를 가져와서 GlobalAveragePooling2D()를 통해 차원을 1차원으로 축소한다. 그리고 Dense()를 통해 입력과 출력을 연결하여 fully connected layer를 만든다. 이 때 dropout으로 과적합을 방지하여 학습시킨다

**\*predict\_class에서 중심적으로 봐야 할 부분**

학습된 model과 새로운 이미지를 input으로 받아 이미지를 예측한다.

.txt 파일을 이용해 미리 각 음식에 대한 레시피와 url 데이터를 구축하고, 예측한 음식과 일치하는 파일을 출력한다.

만약 예측 음식에 대한 파일이 없으면 레시피가 없다는 알림을 출력한다.