

MINI PROJECT No.2

GT파일의 학습 적합 여부 판단 모델

학습 목적

- Object Detection 모델 생성 시 객체의 크기가 너무 작은 것은 학습에 사용하지 않음
- 학습을 통해 회사에서 보유하고 있는 GT파일이 Object Detection 모델 학습 시 활용 가능한지 판단
- Bounding Box의 크기가 500 이하인 것은 너무 작은 객체로 간주, 학습 제외

사용한 현장 Data

- Object Detection모델 학습에 사용되는 GT파일

GT 파일 내용

- (1) 이미지 내 객체의 클래스 (학습 시 불필요한 정보로 판단)
- (2) Bounding Box의 중심 x좌표
- (3) Bounding Box의 중심 y좌표
- (4) Bounding Box의 너비
- (5) Bounding Box의 높이

학습 Data

- Bounding Box의 중심좌표, 너비, 높이, 학습 적합 여부를 저장한 csv 파일 사용
- 데이터 수 : 115477
- 0 ~ 1 사이로 정규화
- 학습 Data 구성 (학습에 불필요한 정보는 제거 Ex 객체의 클래스)

입력데이터 : Center_X, Center_Y, Width, Height
정답 : Class (0은 학습 불가, 1은 학습 가능)

Center_X	Center_Y	Width	Height	Class
0.865104	0.546296	0.04375	0.040741	0
0.901823	0.511574	0.034896	0.028704	0
0.831771	0.472685	0.030208	0.021296	0
0.578646	0.338889	0.047917	0.188889	1
0.740625	0.339815	0.144792	0.614815	1
0.327083	0.36713	0.042708	0.176852	1
0.927865	0.372685	0.025521	0.126852	1

학습 모델 - Keras

라이브러리 import

```
import pandas as pd
from sklearn.model_selection import train_test_split
from tensorflow.keras.utils import plot_model
import numpy as np
from tensorflow import keras
```

Data 로드 및 Trainset Testset 생성

```
# 데이터 로드
DATA = pd.read_csv('0507.csv', encoding='utf-8')

# 데이터 확인
#print(DATA.head())

# Train 및 Test Dataset 생성
data = DATA.drop('Class', axis=1).values
target = DATA['Class'].values
x_train, x_test, y_train, y_test = train_test_split(data, target, test_size=0.3, stratify=target, random_state=0)
```

학습 모델 - Keras

Keras 모델 (layer 64개, epoch 20)

```
# 모델 함수 생성
def create_model():
    model = keras.Sequential([
        keras.layers.Dense(64, activation='relu'),
        keras.layers.Dense(len(y_train), activation='softmax')
    ])

    model.compile(optimizer='adam',
                  loss='sparse_categorical_crossentropy',
                  metrics=['accuracy'])

    return model
```

학습 및 테스트 진행

```
# Trainset을 이용하여 학습
model = create_model()
model.fit(x_train, y_train, epochs=20)

# Testset을 이용하여 accuracy 계산
test_loss, test_acc = model.evaluate(x_test, y_test, verbose=2)
print('\ntest accuracy:', test_acc)
```