

CNN 기반의 이진분류 예측 모델

20191048 정보통계학과 박윤희

문제 정의

◆ 문제 정의

Baseball vs Cricket classification

◆ 문제 선정 배경



Baseball과 Cricket이 유사하게 생긴 스포츠이다.

1. 이처럼 유사하게 생긴 이미지를 CNN 모델이 잘 분류할 수 있는가?
2. CNN모델과 전이학습을 사용한 모델의 차이점
3. 분류가 잘 되었다면, 설명가능 인공지능으로 어느 부분을 보고 해당 이미지를 분류하였는가? (가설: 배트와 공)

이렇게 세가지에 대한 답을 구하기 위해 해당 문제를 선정하였다.

문제 정의

◆ 현재 적용 가능한 예측 모델

Sports classification

◆ 완료 후 공헌

현재 있는 모델은 더 다양한 스포츠를 분류하는 모델이다.
하지만, 설명가능 인공지능을 사용하지 않았고, 전이학습을 이용하였지만 이용하기 전과 후의 비교가 없다.
따라서 유사한 스포츠 2가지만 가지고, 설명가능 인공지능의 결과와 전이학습의 이점을 이를 통해 밝히고자 한다.

Data Description

◆ 자료 수집

- 자료 수집 방법: <http://www.Kaggle.com>의 스포츠 데이터 셋 중 baseball과 cricket 데이터 셋을 다운받음.
- 총 1442장의 이미지 데이터
- 2가지 class ['baseball', 'cricket']



baseball



cricket

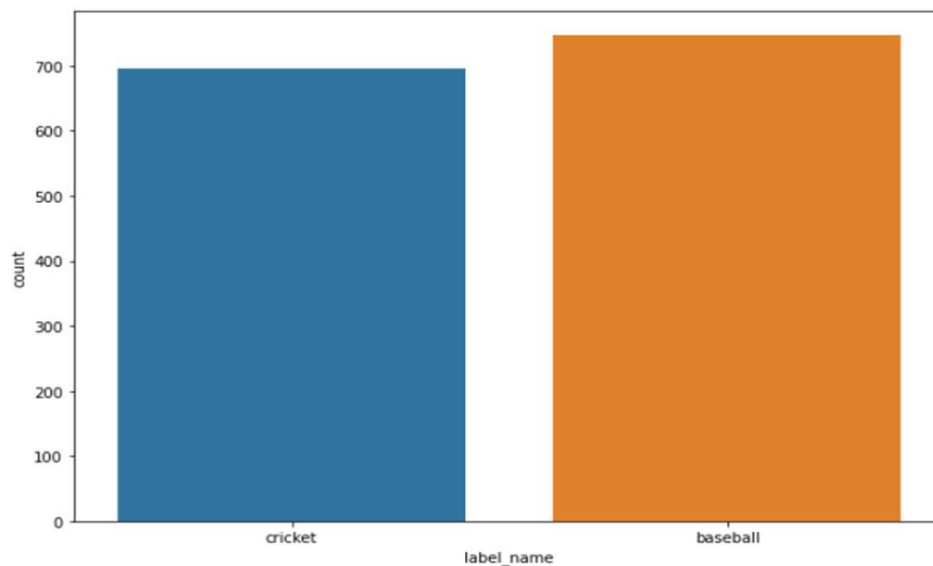
Data Description

◆ 자료 탐색

Examples of dataset sorting by label



Label ratio



Cricket = 696장(48.27%)
Baseball = 746장(51.73%)
→ 데이터 불균형은 문제되지 않음.

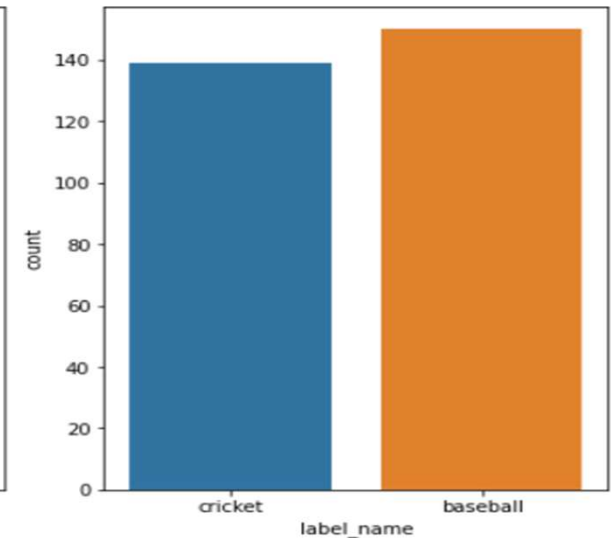
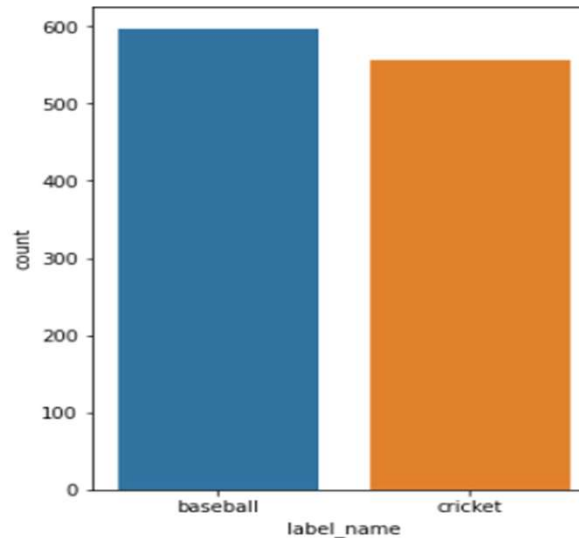
Data Description

◆ Dataset

train_test_split

- stratify 사용
- test_size = 0.2

Baseball: 596장 & 150장
Cricket: 557장 & 139장



ImageDataGenerator

- ImageAugmentation 사용
- 최대최소 정규화 사용
- Target_size = (150,150)
- Batch_size = 16

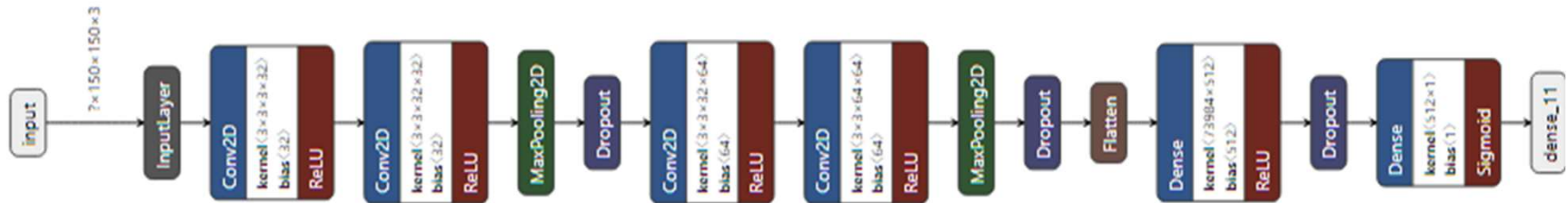


Found 1109 validated image filenames belonging to 2 classes.
Found 278 validated image filenames belonging to 2 classes.

Model Description

◆ cnn1

네트워크 구조

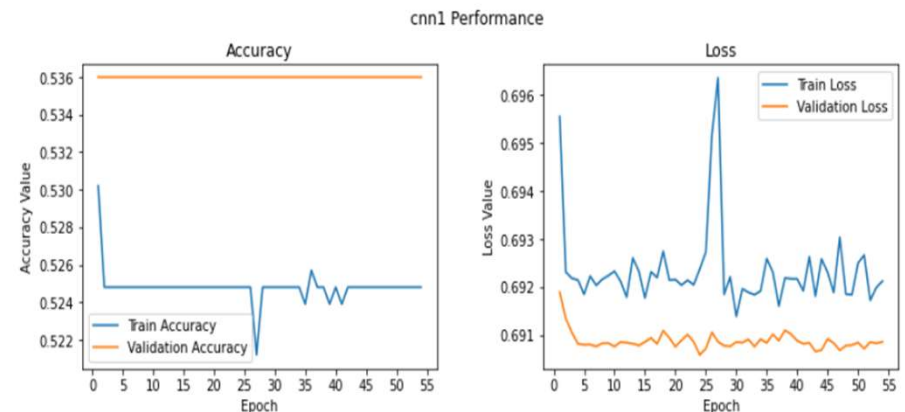


<https://netron.app/>

하이퍼 파라미터

- Optimizer = Adam()
- Dropout 사용
- n_epoch = 100
- early_stopping (patience = 30)

성능



Model Description

◆ cnn2 - final

네트워크 구조



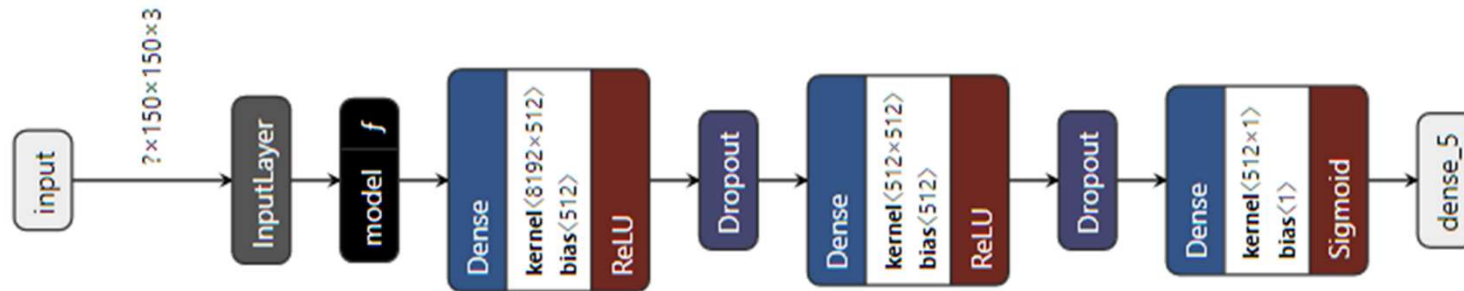
하이퍼 파라미터

- Optimizer = Adam()
- n_epoch = 1024
- early_stopping = 30
- reduce_lr = ReduceLROnPlateau (monitor='val_loss', factor=0.5, patience=6, min_lr=0.000001)
- class_weights = "balanced"

Model Description

◆ TL_vgg1 – final

네트워크 구조



<https://netron.app/>

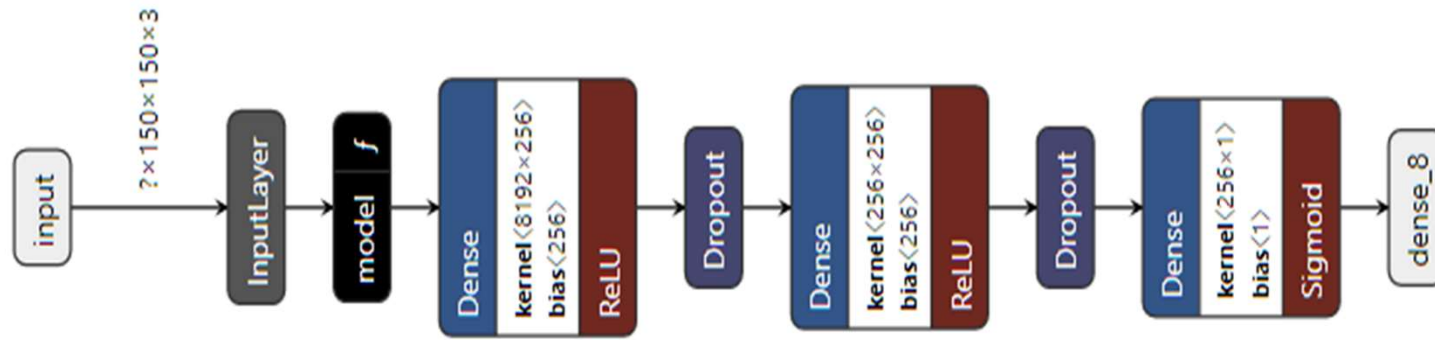
하이퍼 파라미터

- Pre-trained model = vgg16
- Optimizer = Adam()
- Dropout 사용
- n_epoch = 1024
- early_stopping (patience = 30)

Model Description

◆ TL_resnet1 – final

네트워크 구조



<https://netron.app/>

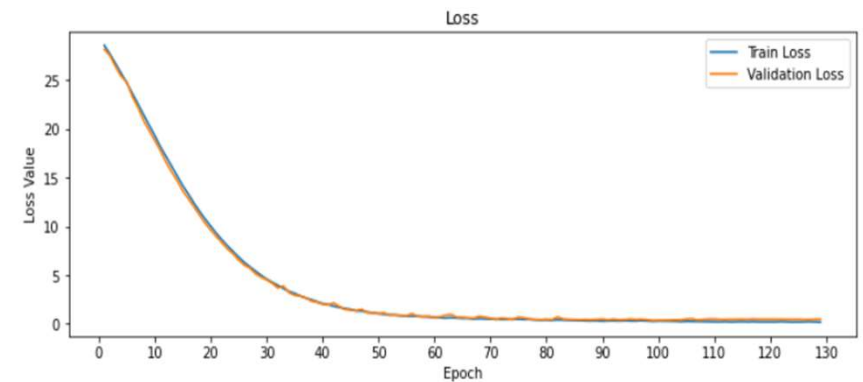
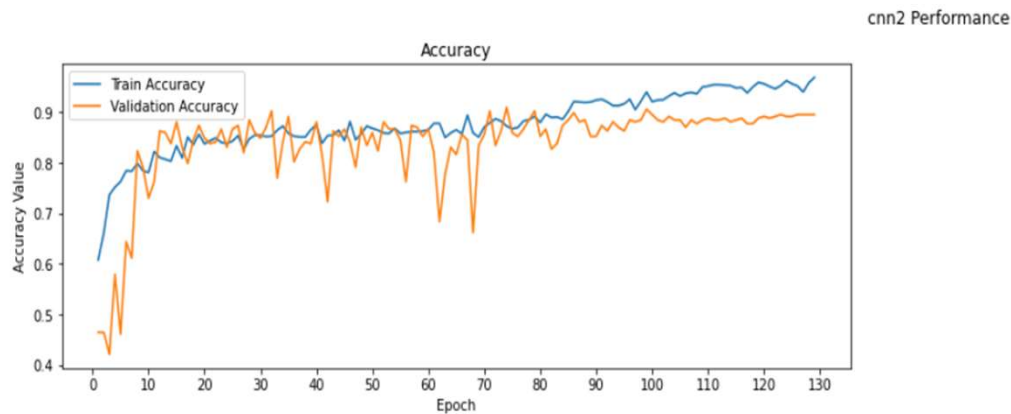
하이퍼 파라미터

- Pre-trained model = resnet50
- Optimizer = Adam()
- Dropout 사용
- n_epoch = 1024
- early_stopping (patience = 30)

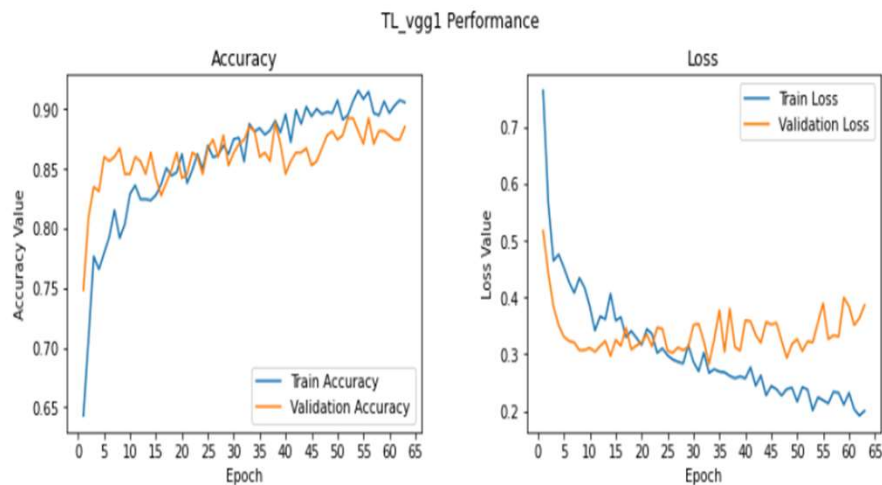
Experimental Results

◆ 최종모델 성능

cnn2



TL_vgg1

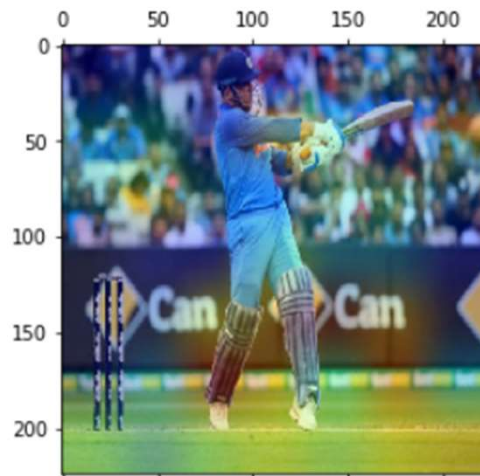
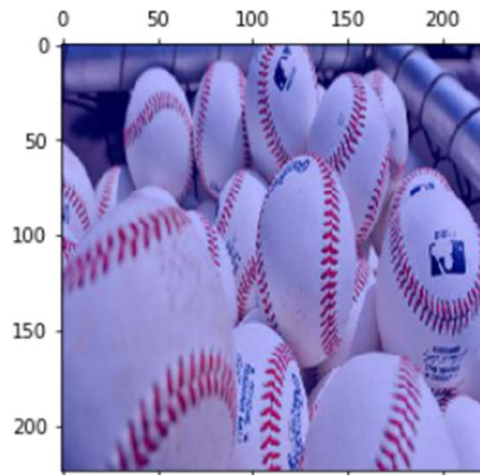


TL_resnet1

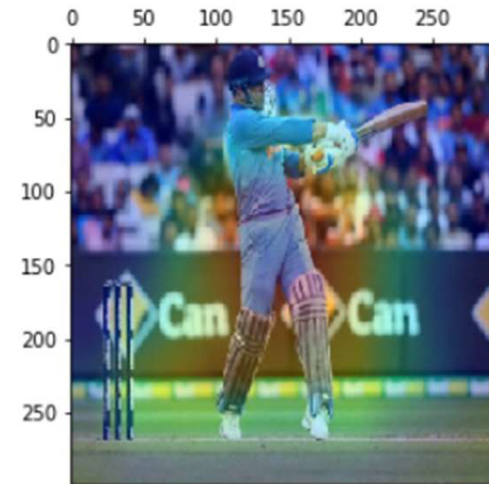
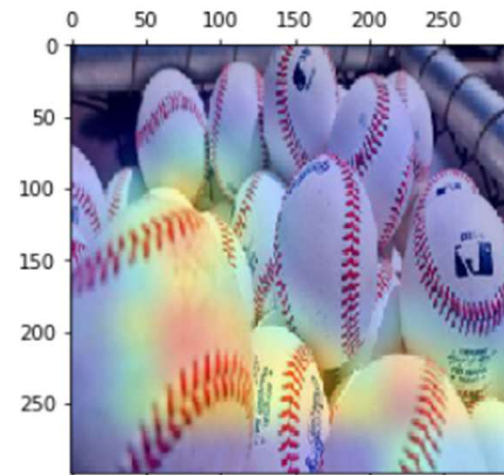


◆ gradCAM

resnet50



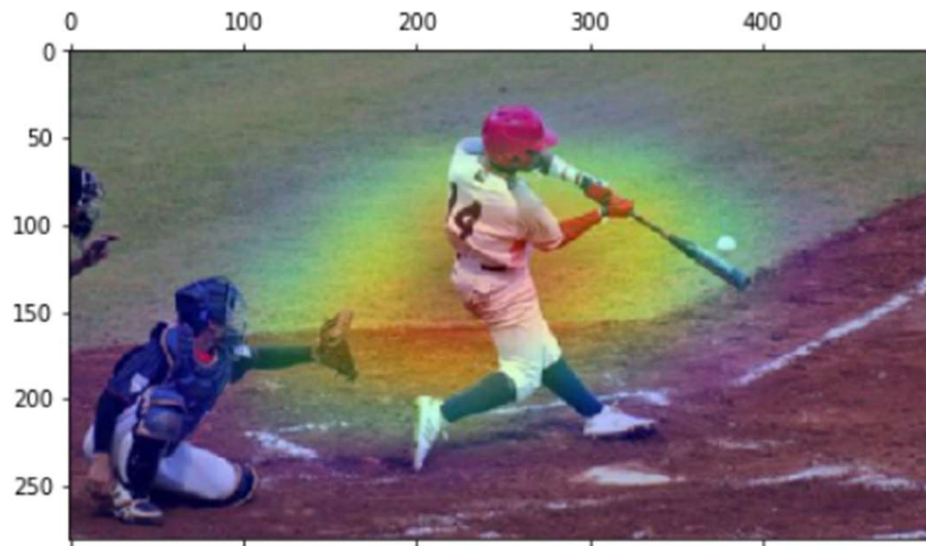
xception



◆ gradCAM

xception

Resnet50과 xception을 해본 결과 xception이 더 잘 나왔다고 판단해 야구에 대해 다시 한번 해봄.



▶ 결과적으로 유니폼과 포수쪽을 보고 cricket과 baseball을 분류한 것으로 보임.

Discussion

◆ 결론

1. 배트 모양으로 판단할 것이라고 생각했지만, 포수 쪽이나 유니폼을 보고 판단함.
2. 어떤 pre-trained model이든, 커스텀한 cnn모델이든 성능이 꽤 잘나옴.

◆ 궁금한점(아쉬운점)

1. 나의 모델에 XAI(gradCAM)을 삽입하는 것이 안됨.
2. 학습은 매우 잘되었지만, 모든 모델의 roc_curve가 잘 안나옴. prediction부분에 문제가 있다고 판단됨.

