

# 너.. 약간 포켓몬 답은 것 같은데?

응용수학과 김수현  
응용수학과 박승혜  
응용수학과 서민정  
응용수학과 이동주



# 모델 학습 데이터 정제



- 1. Crawling한 이미지 중 학습할 데이터 선정
  - Crawling 해온 이미지들은 순차적으로 긁어온 데이터로, 불완전한 데이터가 포함된다. 모델의 정확성을 높이기 위해 학습에 사용할 데이터를 선택하며, Crawling으로 가져온 사진 중 해당 포켓몬과 일치하는 사진 20장을 선정한다.
- 2. 이미지 resizing
  - 모델 학습에 사용할 이미지는 라벨링 전 같은 크기로 만들어준다. 내 컴퓨터에 저장된 파일을 불러와 원하는 크기로 resize한 후 저장한다.

## 이미지 처리를 위해 PIL 라이브러리를 사용

```
1 import glob
2 from PIL import Image
3
4 # jpg 파일명 경로 받아 읽기
5 imglist_jpg = glob.glob('C:/Users/82109/Desktop/PythonWorkspace/khuthon/피카츄/*.jpg')
6 # png 파일명 경로 받아 읽기
7 imglist_png = glob.glob('C:/Users/82109/Desktop/PythonWorkspace/khuthon/피카츄/*.png')
```

이미지가 저장된 위치에서 해당 확장자명을  
갖는 이미지별로 전부 읽어 리스트에 저장

## Resize 크기 설정

```
10 # 가로, 세로 변환 크기 설정
11 width = 150
12 height = 150
13
14 # jpg 파일 변환
15 for img_path in imglist_jpg:
16     img_jpg = Image.open(img_path)
17     resize_img = img_jpg.resize((width, height), Image.ANTIALIAS)
18     resize_img.save(img_path)
19
20 # png 파일 변환
21 for img_path in imglist_png:
22     img_png = Image.open(img_path)
23     resize_img = img_png.resize((width, height), Image.ANTIALIAS)
24     resize_img.save(img_path)
```

리스트에 저장된 이미지들을 resize한 뒤 저장

# Modeling part(1)



- 1. 포켓몬 학습 모델
  - : 각 포켓몬 마다 사진 20장을 준비하고 해당 사진을 array 형식으로 형 변환 시킨 뒤 해당 포켓몬에 해당하는 이름을 label로 주고 학습시킨다. Label은 string 형식이므로 one-hot encoding 을 했고 총 30가지 종류의 포켓몬 카테고리가 있다.
- 2. 같은 카테고리 내부의 사진 학습 모델
  - : 위 모델이 그 사람과 가장 닮은 포켓몬을 학습한 모델이라면 , 두 번째 모델은 20가지의 해당 포켓몬 이미지 중 어떤 이미지와 가장 닮았는지를 학습한다. 각 포켓몬 카테고리마다 20장의 사진으로 하나의 모델을 학습한 후 위의 모델의 결과를 토대로 가장 닮은 이미지를 제공한다.

```
In [54]: MODEL_DIR = './model/'
if not os.path.exists(MODEL_DIR):
    os.mkdir(MODEL_DIR)
model_path = './model/{epoch:02d} - {val_loss:.2f}.hdf5'
checkpointer = ModelCheckpoint(filepath=model_path, monitor='val_loss', verbose=1, save_best_only=True)
early_stopping_callback = EarlyStopping(monitor='val_loss', patience=7)
```

```
model=Sequential()
model.add(Conv2D(32, kernel_size=(3,3), input_shape=(150,150,3), activation='relu'))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
```

모델 저장 경로를 설정한다.

```
model.add(Conv2D(32, (3,3)))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
```

```
model.add(Conv2D(64, (3,3)))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
```

```
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Activation('relu'))
model.add(Dropout(0.5))
model.add(Activation('relu'))
model.add(Dense(29, activation='softmax'))
```

```
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
history = model.fit(attr, label, validation_data=(attr, label), epochs=50, batch_size=145, verbose=1, callbacks=[early_stopping_callback, check
```







# 웹 개발



Framework: Django

## 구현한 기능

1. 직접 파일 업로드
2. 또는 웹캠을 이용해 직접 촬영 가능
3. 업로드된 파일을 이미 학습시켜놓은 모델에 넘겨 어떤 포켓몬과 유사한지 판단
4. 가장 유사도가 높은 포켓몬의 이미지와 사용자의 이미지를 함께 페이지에 전달





구현 결과

나와 닮은 포켓몬은?

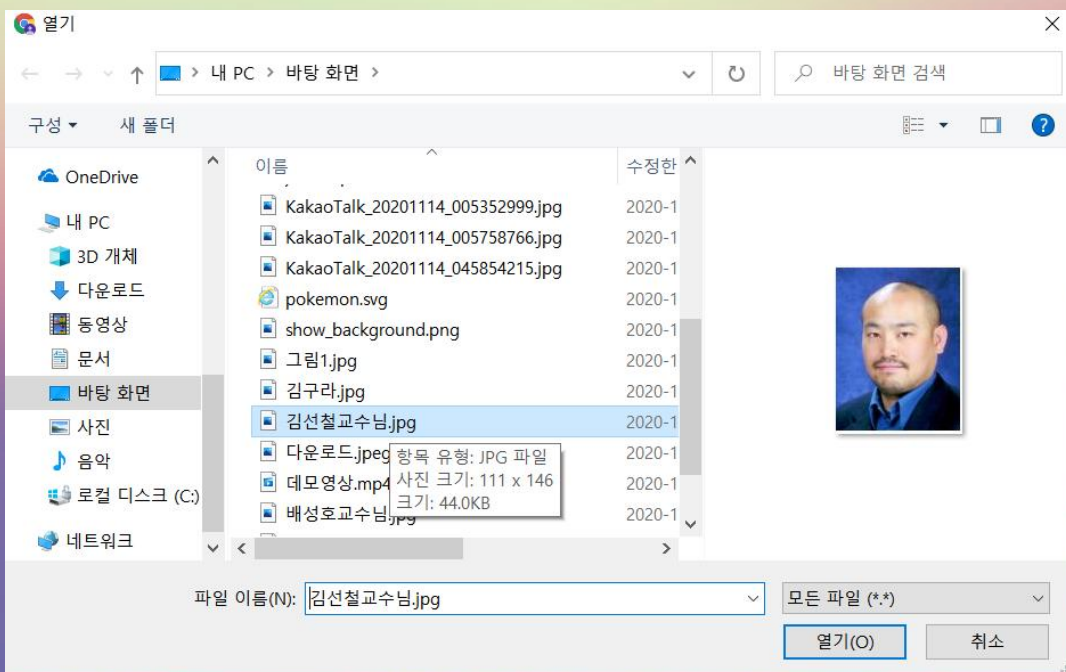
메인 페이지

너.. 약간 그 포켓몬 닮은 것 같은데?



Developed By Bulgari

# 파일 업로드



게임PX디자인 Matplotlib tutorial

업로드하세요.

파일 선택

선택된 파일 없음

파일 선택

업로드

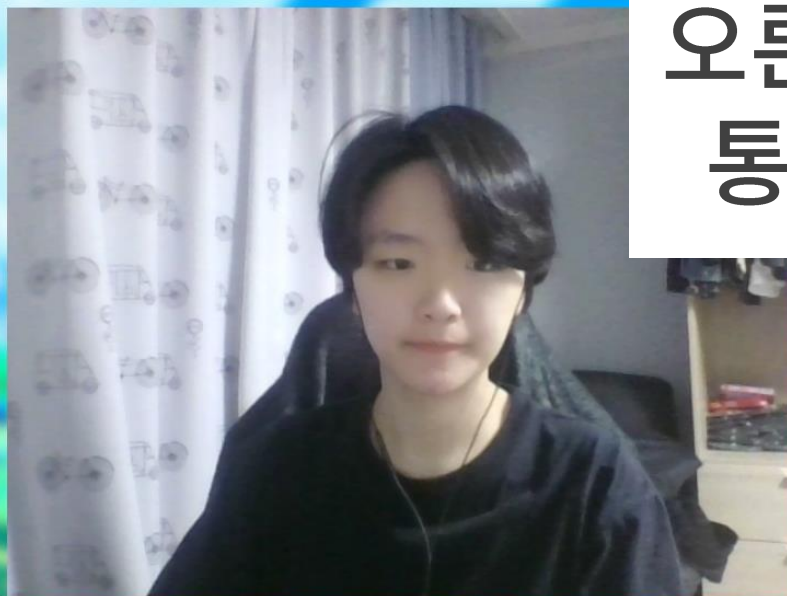


# 파일 업로드 후 결과 페이지

아, 그래! 넌 디그다 (을)를 60.13001799583435%만큼 닳았어!



나와 닮은 포켓몬은?



웹캠으로 직접 촬영  
왼쪽은 실제 웹 캠을  
통해 보여지는 video  
오른쪽은 촬영 버튼을  
통해 촬영한 이미지

결과: 롱스톤 56%

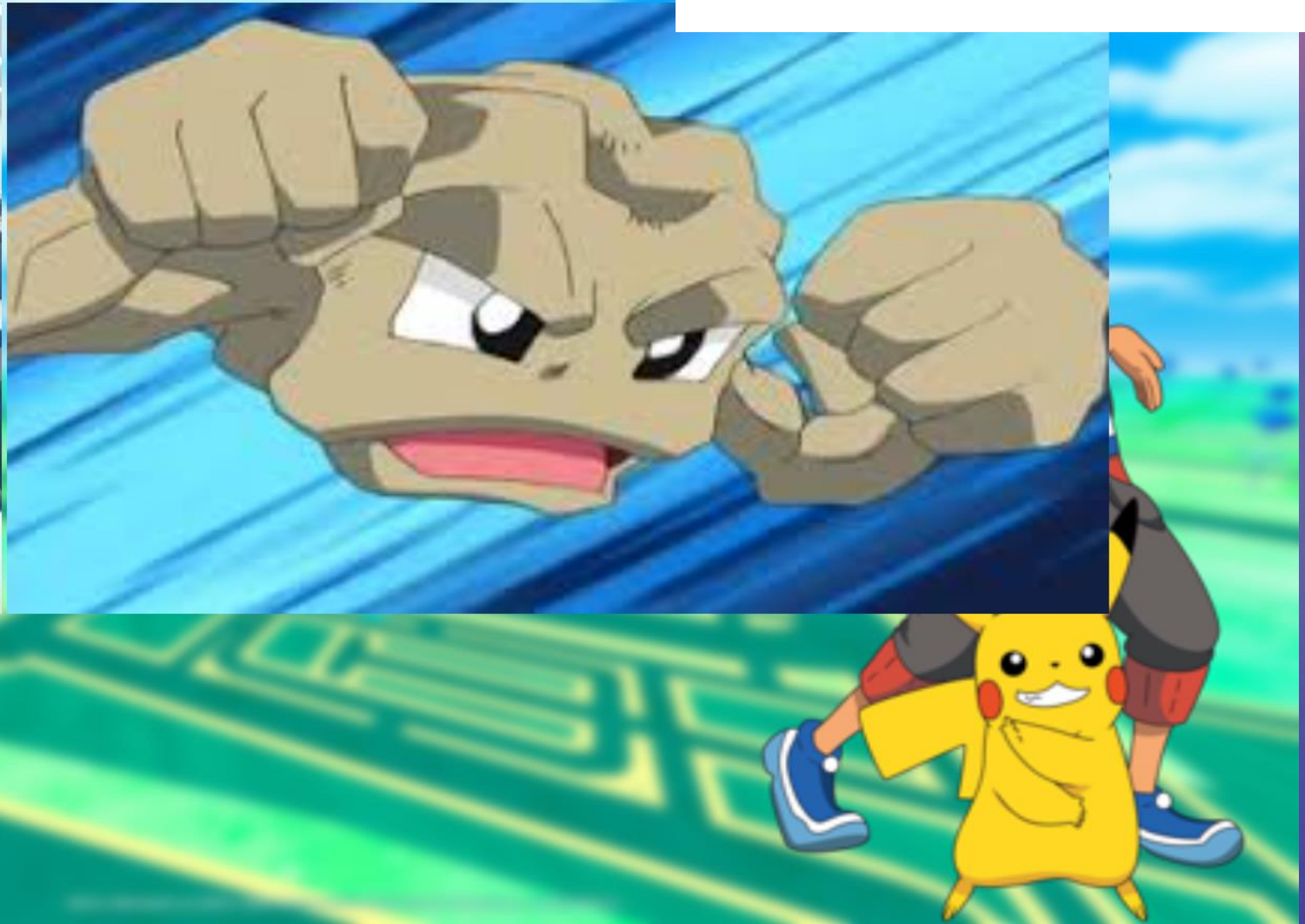
아, 그래! 넌 롱스톤 (을)를 56.32670521736145%만큼 닮았어!





결과: 꼬마돌 45%

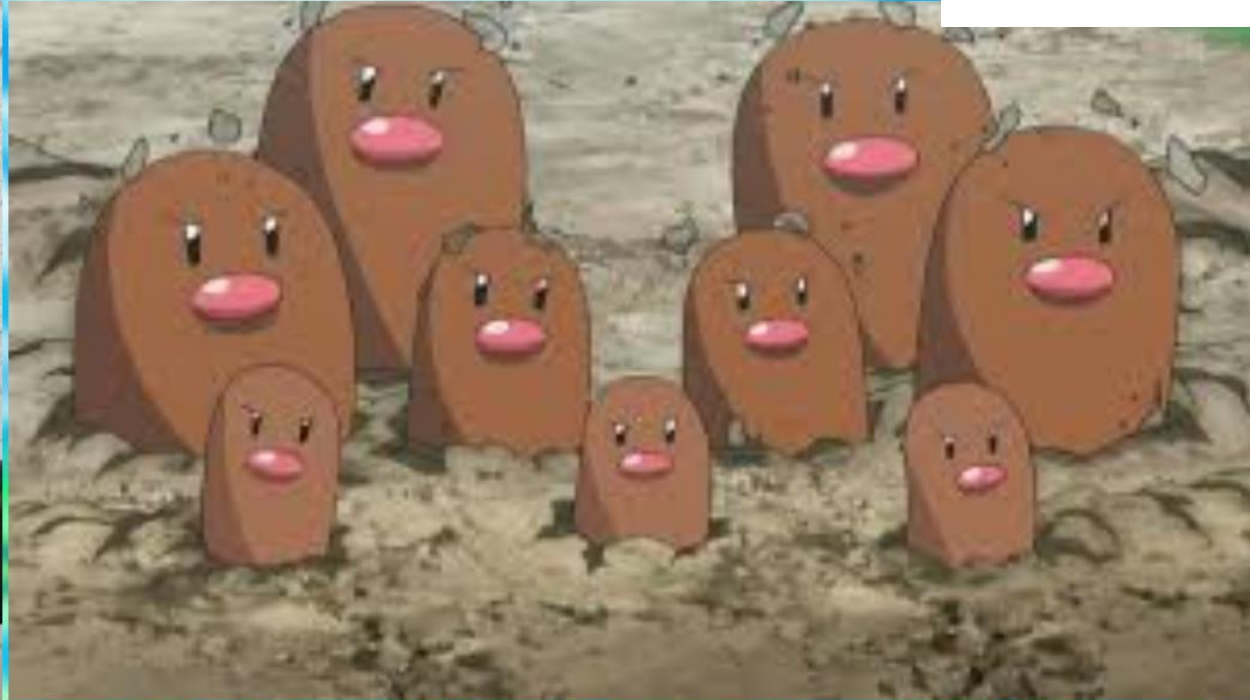
아, 그래! 넌 꼬마돌 (을)를 45.50596475601196%만큼 닮았어!





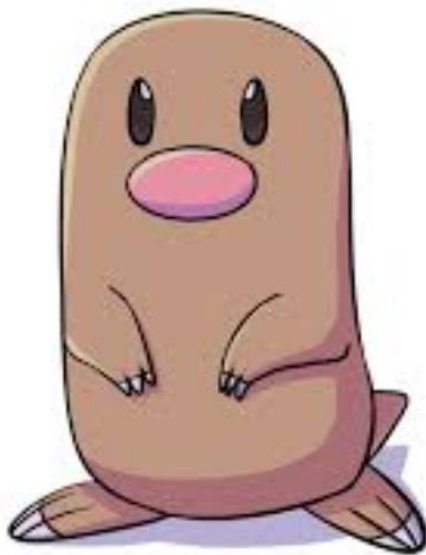
결과: 디그다 92%

아, 그래! 넌 디그다 (을)를 92.75273084640503%만큼 닮았어!



결과: 디그다 60%

아, 그래! 넌 디그다 (을)를 60.649627447128296%만큼 닮았어!





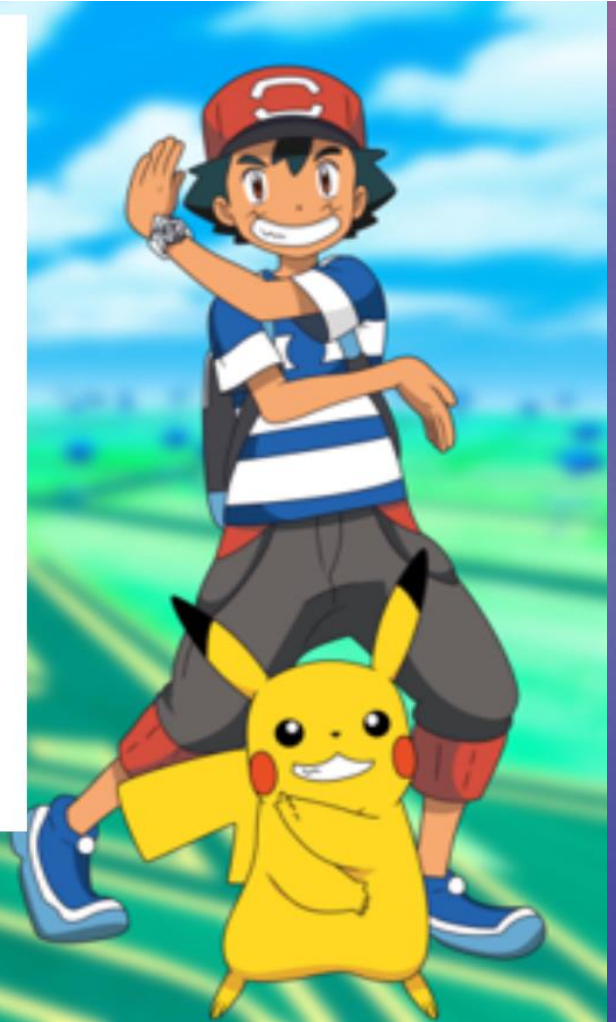
결과: 토게피 49%

아, 그래! 넌 토게피 (을)를 49.155157804489136%만큼 닮았어!



결과: 디그다 93%

아, 그래! 넌 디그다 (을)를 93.4058666229248%만큼 닮았어!





Thank you