

**KU-BIG 홈커밍**

# **Object Detection**

권지혜. 이은진. 명재성  
구형석. 유정아. 원혜진

# 프로젝트 소개



인스타그램 피드  
object detection을  
통한,

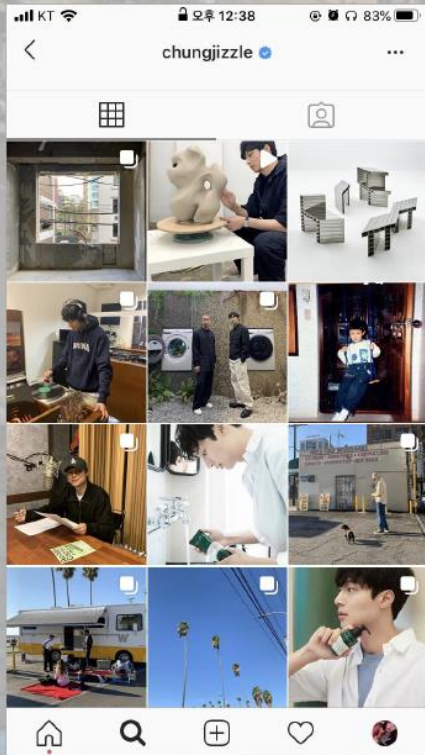
유저에게 어울리는  
음악 추천

#꿀잼\_예상  
#딤러닝  
#크롤링

2020.04.23



# 예상 결과물



당신의  
Instagram  
ID를 입력해주세요

chungjizzle

당신과 어울리는 음악은? **GO**

당신은  
#예술파  
이시군요!

당신과 어울리는 음악은?



Artist  
지코 (ZICO)



Dali, Van, Picasso  
빈지노 (Beenzino)

멜론으로 듣기

재생하기



# 진행 방향

# 진행 방향

---

이미지  
크롤링



Object  
Detection



추출된  
Label  
유형화



음악 추천  
웹 구축

# 이미지 크롤링 예시

무엇을 하게 될까요?

상세 내용

## 1. 데이터 크롤링

- 1) Selenium을 활용하여  
Instagram crawler를 생성합니다.



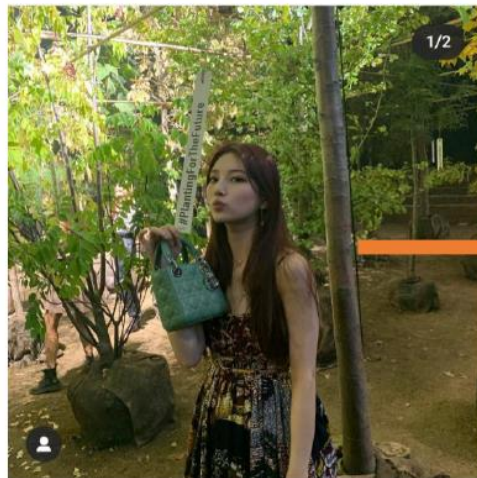
2) 크롤링 할 대상은,



skuukzky  
파리

:

장소



사진



~~XXXXXXXXXX~~ 님 외 여러 명이 좋아합니다

skuukzky #PlantingForTheFuture  
@dior

해시태그





```

from selenium.webdriver.chrome.options import Options
from selenium.common.exceptions import NoSuchElementException

import urllib.parse
import urllib.request
import time
import datetime
import csv

```

```

def scroll_down(webdriver):
    webdriver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
    time.sleep(3)

def add_photo(webdriver):
    temp_list = []
    one_photo = webdriver.find_elements_by_class_name("FFVAD")
    for n in one_photo:
        temp = {}
        temp['alt'] = n.get_attribute('alt')
        temp['src'] = n.get_attribute('src')
        temp_list.append(temp)
    return temp_list
#넘어오는 형식은 [{alt:내용, src:주소},{alt:내용, src:주소}]

#검색을 원하는 insta ID
insta_id = input("Input Your Insta ID : ")
url = "https://www.instagram.com/" + insta_id

webdriver = Chrome('C:/Users/saki7/chromedriver.exe')
webdriver.get(url)
session = webdriver.requests_session
response = session.get(url)

#포스트의 총 개수
len_post = webdriver.find_element_by_class_name('g47SY').text

photo_list = []
try:
    while True:
        for n in add_photo(webdriver):
            if n in photo_list:
                pass
            else:
                photo_list.append(n)

        scroll_down(webdriver)

        if (int(len_post) == len(photo_list)):
            break
except:
    pass
for i, n in enumerate(photo_list):
    urllib.request.urlretrieve(n['src'],str(i)+''.jpg')

```

## Input Your Insta ID : dlwlrma

```

{'alt': 'Photo by 이지금 on May 06, 2020. 이미지: 사람 1명',
 'src': 'https://instagram.ficn6-1.fna.fbcdn.net/v/t51.2885-15/sh0.08/e35/s640x640/95568948_167961974580082_5631483226254233821_n.jpg?_nc_ht=instagram.ficn6-1.fna.fbcdn.net&_nc_cat=1&_nc_ohc=xad0WZnB7YsAX-I0x9V&oh=036c286c49f6690227c0758c87e9477a&oe=5EE53399'},
{'alt': 'Photo by 이지금 on May 05, 2020. 이미지: 사람 1명, 근점 촬영',
 'src': 'https://instagram.ficn6-1.fna.fbcdn.net/v/t51.2885-15/sh0.08/e35/s640x640/95862917_159376382232187_2379713728898820327_n.jpg?_nc_ht=instagram.ficn6-1.fna.fbcdn.net&_nc_cat=1&_nc_ohc=JUaPepdg0J8AX-DFfwC&oh=277793407e0227a6b9df366b9c578772&oe=5EE5FA68'},

```



# Object Detection 예시

In [1]:

```
# !pip install --upgrade google-cloud-vision
```

In [1]:

```
def detect_labels_uri(uri):
    from google.cloud import vision
    client = vision.ImageAnnotatorClient()
    image = vision.types.Image()
    image.source.image_uri = uri

    response = client.label_detection(image=image)
    labels = response.label_annotations
    print('Labels:')

    for label in labels:
        print(label.description)

    if response.error.message:
        raise Exception(
            '{}\nFor more info on error messages, check: '
            'https://cloud.google.com/apis/design/errors'.format(
                response.error.message))
```

In [12]:

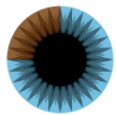
```
detect_labels_uri('https://instagram.ficn6-1.fna.fbcdn.net/v/t51.2885-15/sh0.08/e35/'+
's640x640/92844864_857869931378938_665164534222130907_n.jpg?_nc_ht=instagram.fic'
'n6-1.fna.fbcdn.net&_nc_cat=1&_nc_ohc=PtGsnJTrEAWAX9byh4D&oh='+
'15aebbd8c5b6f35a4b34097b38a02c82&oe=5EE62265')
```



Labels:  
Hair  
Hairstyle  
Lip  
Beauty  
Long hair  
Shoulder  
Jaw  
Brown hair  
Gesture  
Finger

# 딥러닝 기초 스터디

## (1주차)



# 3 blue 1 brown

Introduction example

But what is a Neural Network? | Deep learning, c...  
Neural Networks

From the ground up

YouTube

0:00 / 19:13

Annotations

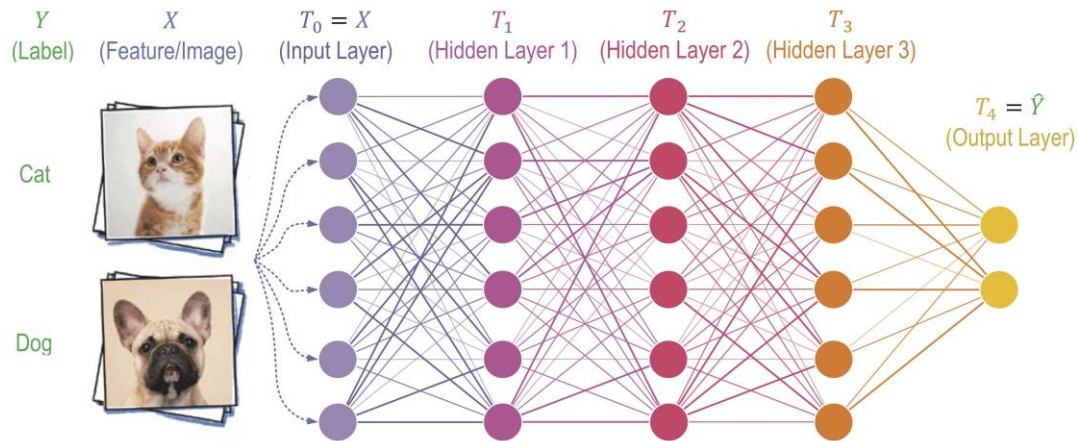
Powered by Annotated

- (0:00) Introduction example
- (1:07) Series preview
- (2:42) What are neurons?
- (3:35) Introducing layers
- (5:31) Why layers?
- (8:38) Edge detection example
- (11:34) Counting weights and biases
- (12:30) How learning relates
- (13:26) Notation and linear algebra

# About Deep Learning

## Neural Network

Neurons. Layers. Weights. Biases



Activation function - sigmoid, ReLU / Cost function

Backpropagation / Gradient Descent

# 이후 스터디

CS231n: Convolutional Neural Networks for Visual Recognition

Spring 2020

Previous Years: [\[Winter 2015\]](#) [\[Winter 2016\]](#) [\[Spring 2017\]](#) [\[Spring 2018\]](#) [\[Spring 2019\]](#)



Lecture 5

Tuesday  
April 21

**Convolutional Neural Networks**

History  
Convolution and pooling  
ConvNets outside vision

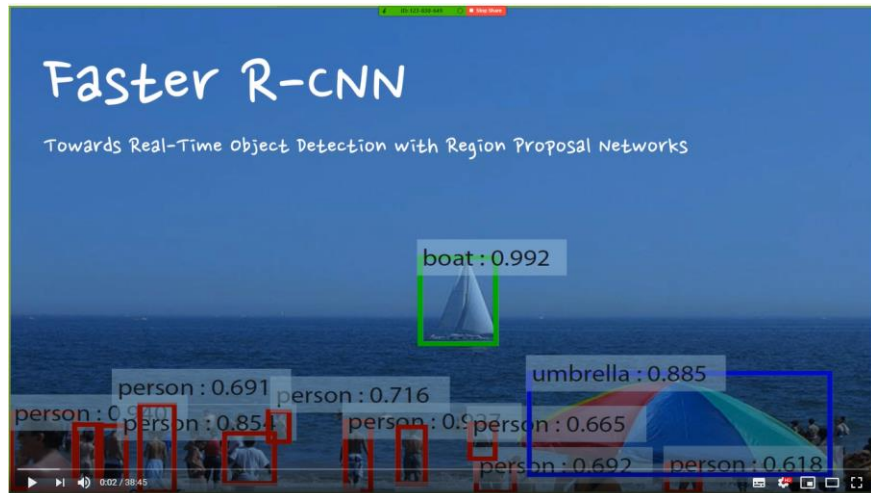
Lecture 9

Tuesday  
May 5

**CNN Architectures**

AlexNet, VGG, GoogLeNet, ResNet, etc

## Faster R-CNN



PR-012: Faster R-CNN : Towards Real-Time Object Detection with Region Proposal Networks

# 이후 일정

---

5/18~5/23	CNN, Faster R-CNN 스터디
5/23~5/30	크롤링, Object Detection 구현
5/31~6/06	Labeling된 데이터 유형화 작업
6/07~6/10	음악 추천 웹서비스 구현



# 고민 사항

# 고민 사항

---

## 〈Object Detection〉

### 1. 직접 구현

- Faster R-CNN을 일단부터

### 2. 기존 API 사용

Keras	코드에 대한 공부는 진행	Label 80~90개 정도
Google Vision		Label 1900개 정도

# 고민 사항

---

## 〈Label 유형화〉

Object Detection 결과로 나온  
Label들을 어떻게 유형화할 것인가?

Label들을 vectorization 해서  
가장 거리가 가까운 유형과 연결?  
등등 고민 중입니다.

## 〈음악 추천〉

유형별로 어떤 음악을 추천할 것인가?

유형별로 미리 음악을 설정해 놓을지,  
새로운 음악 추천 알고리즘을 찾아볼지,  
등등 고민 중입니다.

---

# 감사합니다