



프로젝트 진행상황 - day01

BE<sub>12</sub>

유승은 임소연 전주영

# Prepare Data

1. 단개체 학습 데이터(Single Object Detection)에 적절한 데이터만 골라내기
  - > 수작업
2. 너무 큰 사이즈의 데이터
  - > 데이터 리사이징 (PIL, Image 라이브러리 사용 - 500x500 이하로 낮추기)
  - > .jpg 파일만 사용
  - > unzip을 통해 데이터 가져오기

# Day 이 목표

1. Single Object Detection 모델 구현
2. classification 목표치 달성
  - 목표 Accuracy: 92%
  - 현재 Accuracy : 83.93%

# Model 수정

1. Overfitting 해결하기
  - > Learning rate Decay 적용하기
  - > Data Augmentation: zoom, width, height, brightness
2. Accuracy 높이기
  - > Learning rate : Decaying 비율 조절
  - > Dense Layer 빼기
3. val Accuracy 높이기
  - > freeze Layer 조절
  - > SGD optimizer 사용 (Learning Rate가 직관적으로 조절되는 모델)
  - > GoogLeNet 모델 구축

# Status - VGG16

VGG16(vanila)	VGG16 + flatten + dense(1024, relu) + BN + dense(10, softmax)								
model_name									
file_name	Ai-kea_vanilla_test1.ipynb		Ai-kea_vanilla_sey.ipynb		Ai-kea_vanilla_jjy.ipynb				
img_size	112 * 112	112 * 112	112 * 112	112 * 112	112 * 112	112 * 112	112 * 112	112 * 112	112 * 112
batch_size	300	200	128	128	128	256	128		
epoch	20	30		50	50	50	100	100	
seed_num	F	F	F	F	F	F	F	F	F
Fine_Tuning	X	X	X	X	X	X	X	X	X
Learning_rate	off	off	off	half-on	on	off	off	on	off
data_aug (option)									
zoom	F	F	F	0.1	0.1	F	F	F	F
width	F	F	F	0.1	0.1	F	0.3	0.3	0.3
height	F	F	F	0.1	0.1	F	0.3	0.3	0.3
brightness	F	F	F	0.8-1.2	0.8-1.2	F	F	F	0.8-1.2
val_acc(max)	<b>79.41</b>	<b>79.85</b>	<b>0.7937</b>	<b>0.7676</b>	<b>0.7850</b>	<b>0.7883</b>	<b>0.7887</b>	<b>0.7546</b>	<b>0.8393</b>
val_acc(final)	<b>78.22</b>	<b>78.55</b>	<b>0.7839</b>	<b>0.7058</b>	<b>0.7676</b>	<b>0.7861</b>	<b>0.7876</b>	<b>0.7383</b>	

# Status - GoogLeNet

img_size	112*112
batch_size	128
epoch	100
seed_num	F
Fine_Tuning	all true dense: off normal: off
Learning_rate	off

zoom	F
width	F
height	F
brightness	F
<b>val_acc(max)</b>	<b>0.8447</b>
<b>val_acc(final)</b>	