

AI Workshop

머신러닝을 여행하는 예술가를 위한 안내서

실습

DADASMIMI

송하윤

목차

1. 지도학습

1-1. 인간지능과 인공지능

1-2. 이미지 분류 - 데이터셋 만들기

1-3. 이미지 분류 - 학습하기

2. 비지도학습

2-1. 강화학습

2-2. Flappy Bird 게임하기

2-3. Flappy Bird 게임 시키기

머신러닝, 흔히 AI 혹은 인공지능이라고 하는 컴퓨터를 학습시키는 방법이 있습니다.

이러한 단어들을 들으면 신기하고 어렵고 원지 모르겠지만 대단한 느낌이 들곤 합니다.

‘머신러닝을 여행하는 예술가를 위한 안내서’ 실습에서 머신러닝의 대표적인 두가지 방법, 지도학습과 비지도학습에 관해 쉽고 간단하게 알아보도록 하겠습니다.

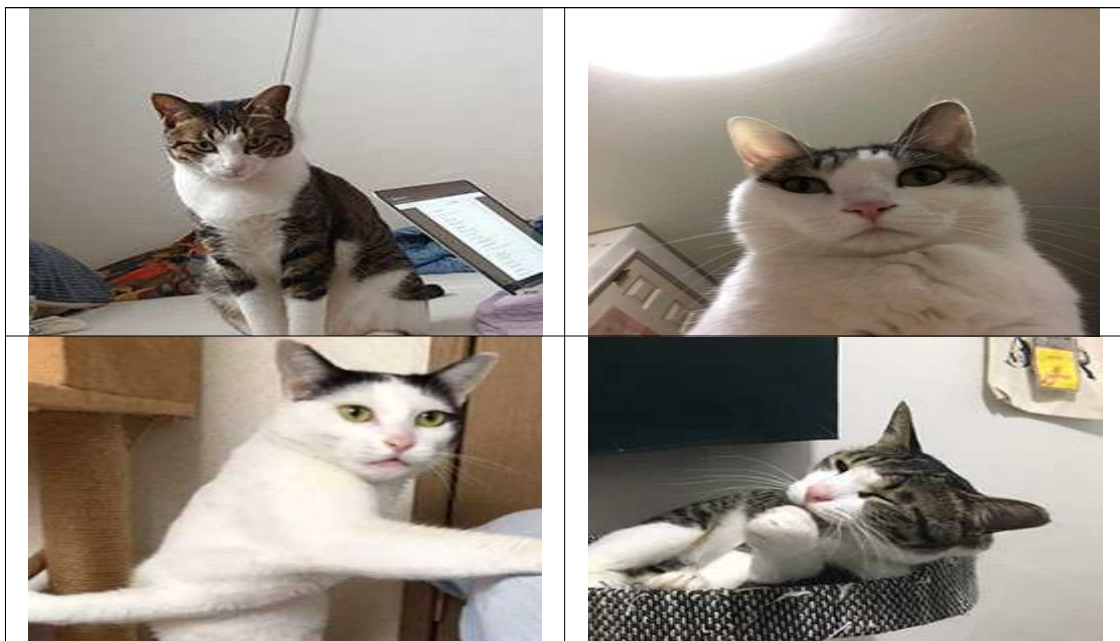
1. 지도학습

지도학습(Supervised Learning)이란, 말 그대로 컴퓨터를 지도하여 학습하는 방법을 말합니다.

정답을 알려주며 학습시키는 학습방법입니다.

제가 기르는 고양이가 한 마리 있습니다. 고양이의 이름은 ‘썬더’ 입니다.

아래의 고양이 사진들(Data)에서 ‘썬더’ 를 찾을 수 있을까요?






1-1. 인간지능과 인공지능

위의 사진만으로는 어떤 고양이가 ‘썸더’ 인지 알 수 없습니다.

이번에는 아래처럼 각 사진에 이름(Label)을 달아보겠습니다.

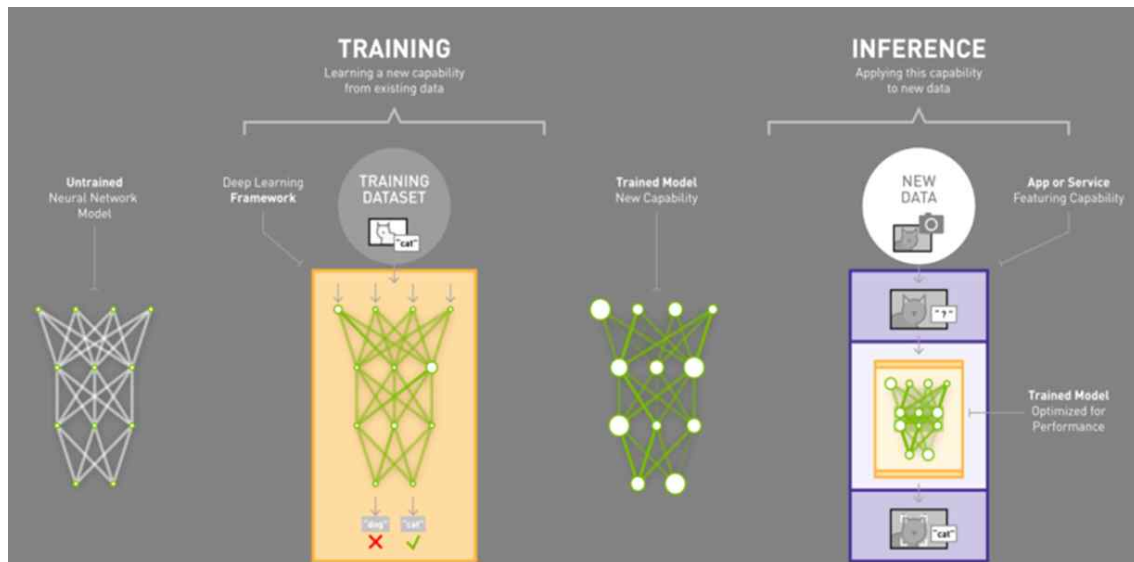
	
Thunder	NotThunder
	
NotThunder	Thunder

이제 아래의 고양이들이 각각 썸더인지 아닌지 알 수 있게 되었습니다.

인간지능(?) 에서는 이름을 붙인 사진들을 한번 학습한 것만으로도 고양이들을 분류 할 수 있습니다.

컴퓨터도 비슷한 방법으로 학습을 합니다.



먼저 학습되지 않은 인공지능의 형태를 만들어 줍니다.

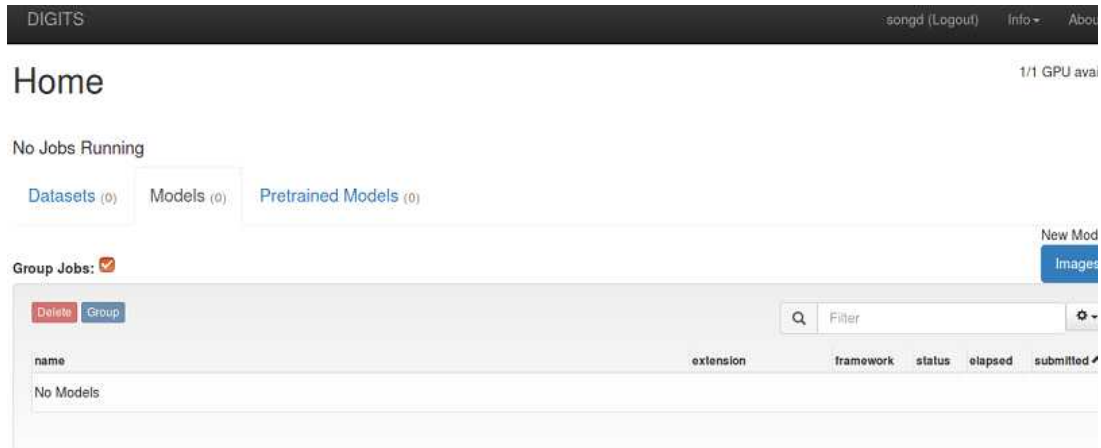
이 인공지능의 형태를 ‘인공지능 모델’ 이라고 합니다.

인공지능 모델에 각각의 사진들(Data)에 이름(Label)을 붙인 것,
데이터셋(Data Set)을 이용해 학습을 시킵니다.

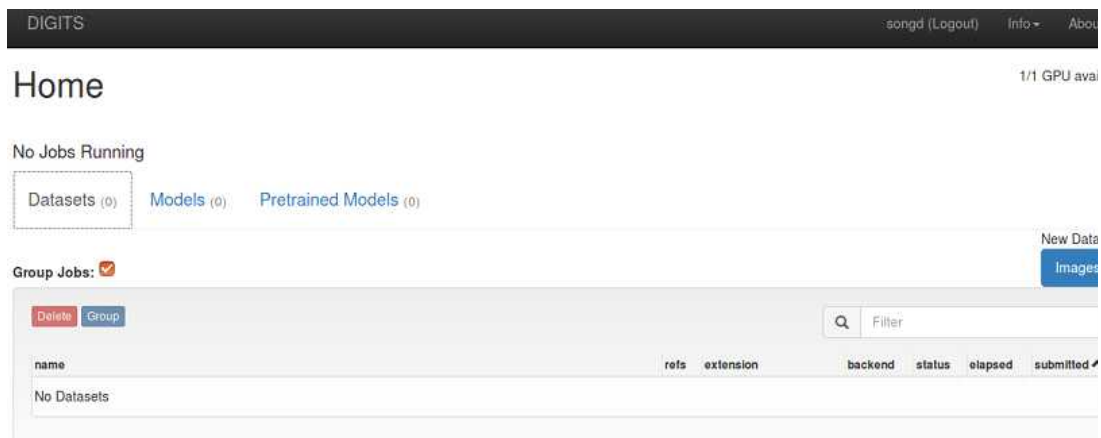
이렇게 각각의 사진들(Data)이 어떤 이름(Label)을 가지고 있는지 먼저 정답을 알려주고 학습시키는 방법이 바로 지도학습(Supervised Learning)입니다.

1-2. 이미지 분류 - 데이터셋 만들기

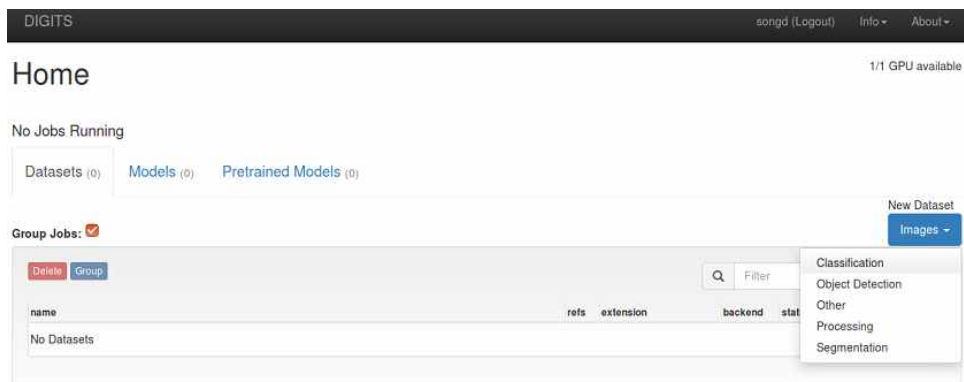
Digits를 실행해 인터넷 창에서 메인 페이지를 띄웁니다.



Datasets 탭을 클릭해 줍니다.



탭이 활성화 되어있는 것을 확인 한 후 오른쪽 Images 버튼을 클릭해 나온 메뉴에서 Classification을 선택 해 줍니다.



로그인을 위한 창이 뜨면 원하는 Username을 입력 후 확인 해 줍니다.

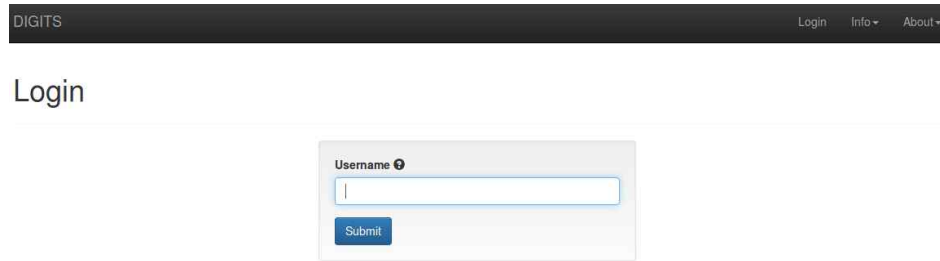
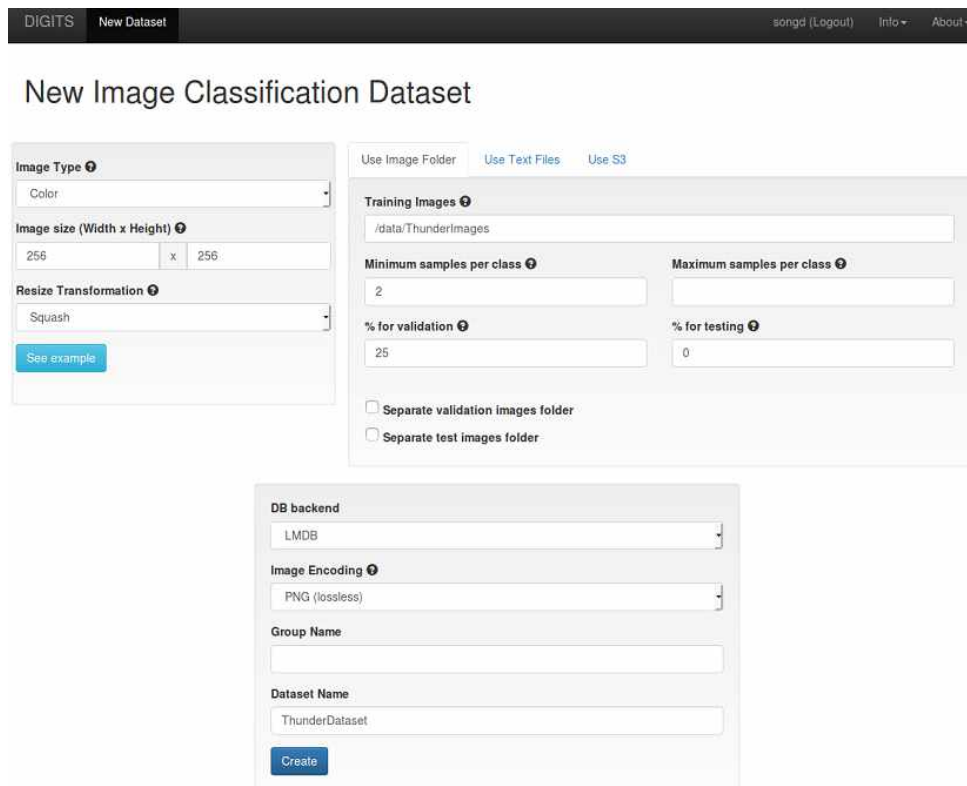


Image Classification Dataset을 만들 수 있는 창이 뜹니다.

다른 항목은 모두 그대로 두고 Training Images에 ThunderImages의 경로를 입력해 줍니다.

가장 아래에 있는 Dataset Name에 원하는 데이터셋 이름을 입력 해 주고 Create버튼을 클릭하여 데이터셋을 생성합니다.



데이터셋을 생성하였습니다!

아래의 사진과 같이 데이터셋이 성공적으로 만들어졌다면 가장 왼쪽 위에 있는 DIGITS 버튼을 클릭하여 메인 페이지로 이동 해 줍니다.

DIGITS

Image Classification Dataset

Logout (Logout)

Help

About

ThunderDataset

Owner: digits

Create Job

Delete Job

Job Information

Job Directory

/home/song/digits/digits/jobs/20180915-233632-a427

Image Dimensions

256x256 (Width x Height)

Image Type

Color

Resize Transformation

Squish

DB Backend

innoDB

Image Encoding

png

DB Compression

none

Dataset size

0 B

Parse Folder (train)

Folder

/data/ThunderImages

Job Status Box

• Initiated at 11:36:02 PM (5 seconds)

• Running at 11:36:04 PM (5 seconds)

• Done at 11:36:07 PM (total=4 seconds)

Parse Folder (train) Done

Create DB (train) Done

Notes

None

Create DB (train)

Input File (before shuffling)

train.txt

DB Creation log file

create_train_db.log

Image Count

8

Categories

10

Image Mean:

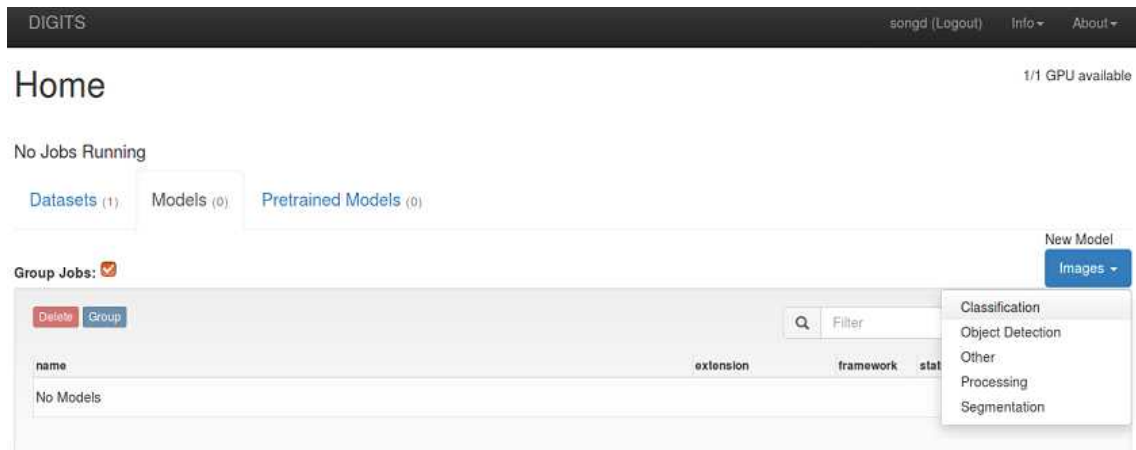


Explore the db

1-3 이미지분류 - 학습하기

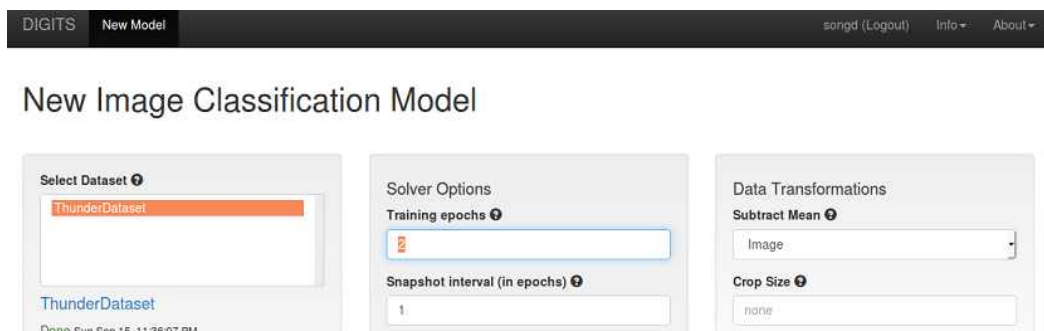
이제 방금 만든 Dataset을 이용하여 이미지 분류 인공지능 모델 (Image Classification Model)을 만들어 보도록 하겠습니다.

아래의 사진과 같이 Models 탭에서 Images버튼을 눌러 Classification을 선택하도록 하겠습니다.

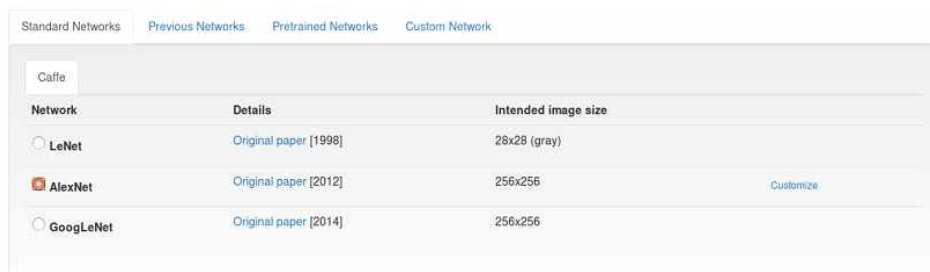


이미지 분류 모델을 만들 수 있는 페이지가 뜨면 방금 전 만들었던 데이터셋을 선택 해줍니다.

Training epochs를 2로 입력 해 줍니다.



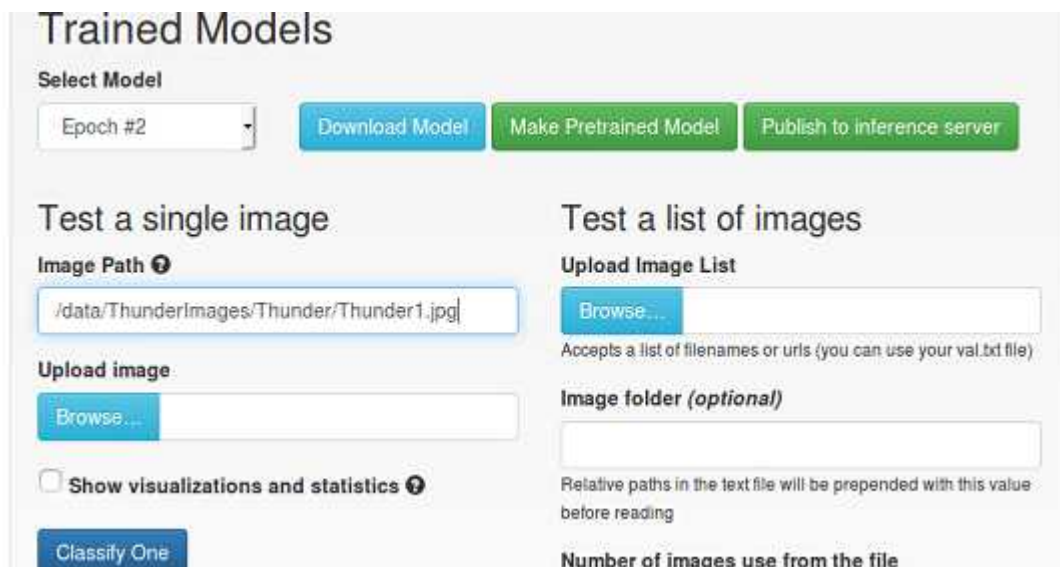
나머지 항목은 그대로 둔 뒤 아래의 Standard Networks 항목에서 AlexNet을 선택해 주도록 하겠습니다.



마지막으로 가장 아래에 있는 Model Name을 원하는 이름으로 입력해 준 후 Create버튼을 눌러 학습시켜보도록 하겠습니다.



학습이 금방 끝납니다. 학습이 끝나면 아래의 사진과 같이 Trained Models 탭에 Image Path 항목에 원하는 이미지의 경로를 입력해줍니다.



아직은 컴퓨터가 두 번의 Epoch동안만 학습했기 때문에 정확하지 않은 분류를 하게 되는 것을 알 수 있습니다.

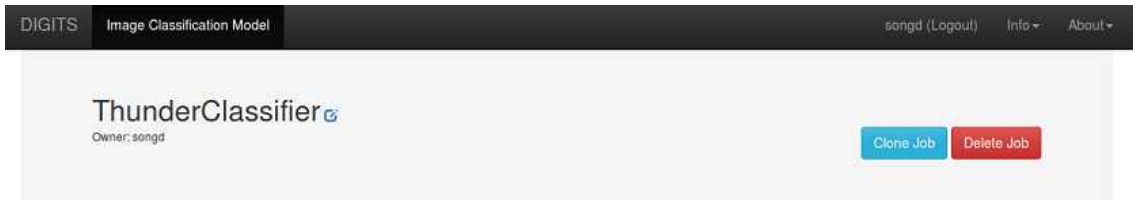
ThunderClassifier Image Classification Model



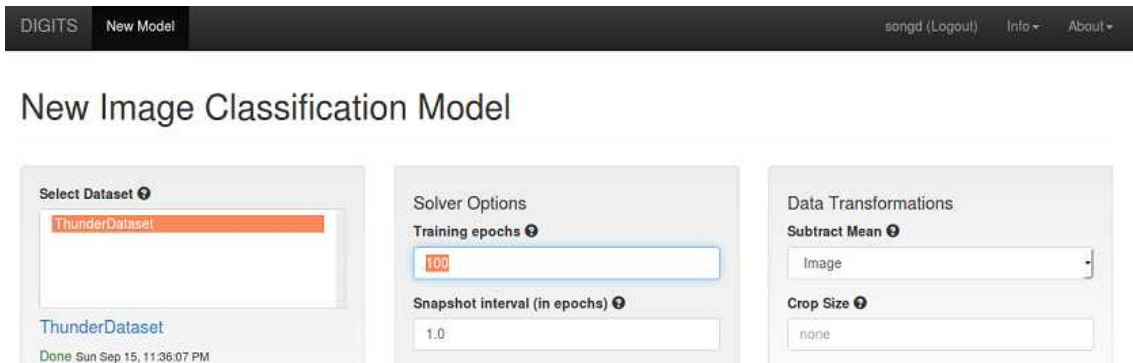
Predictions	
Thunder	51.28%
NotThunder	48.72%

이번에는 100Epoch로 다시 시도해보겠습니다.

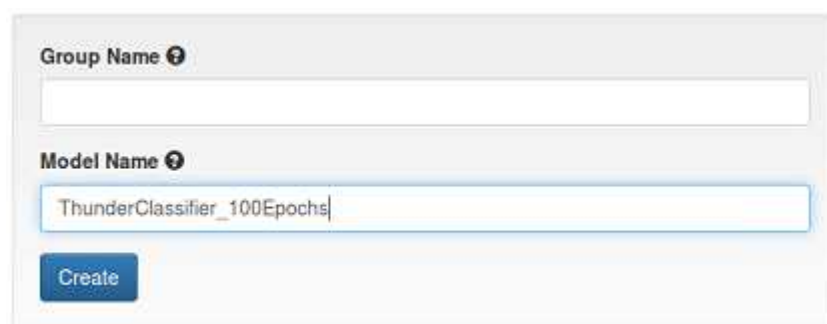
방금 만든 인공지능 모델의 페이지로 돌아가 Clone Job버튼을 클릭해 모델을 복제해 줍니다.



이전에 만든 인공지능 모델과 모든 항목을 똑같이 설정하고 Training Epochs만 100으로 변경합니다.

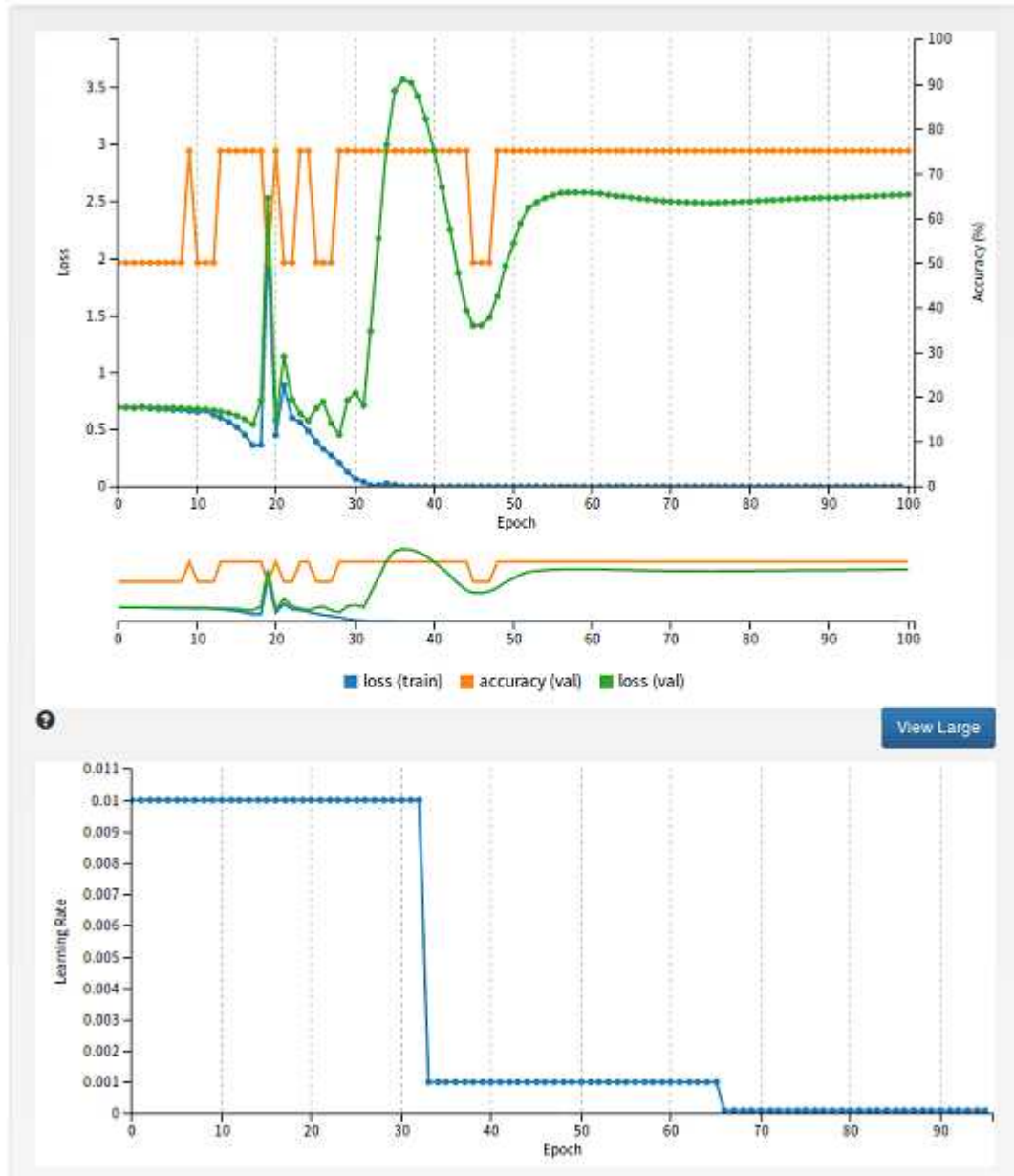


설정이 완료되었다면 Model Name을 입력 후 Create 버튼을 눌러 학습을 시작합니다.



이번 학습은 시간이 좀 걸립니다.

학습이 진행되는 페이지에서 그래프를 확인하며 학습이 잘 진행되는지 확인할 수 있습니다.



학습이 완료되면 Image Path 항목에 원하는 이미지의 경로를 입력한 후 Classift One 버튼을 클릭 해 학습이 어떻게 완료 되었는지 확인 해 보도록 하겠습니다.

100Epoch동안 학습 한 모델은 완전히 정확하거나 매우 정확한 학습 결과를 보여주는 것을 알 수 있습니다.

정답을 알려주고 학습을 시키는 지도학습을 실습해 보았습니다.
위의 과정을 원하는 이미지를 이용해 복습 해 볼 수 있습니다.

2. 비지도학습

비지도학습은 지도학습과는 반대로 정답을 미리 알려주지 않고 학습시키는 방법을 말합니다.

비지도학습의 한 종류인 강화학습을 실습 해 보도록 하겠습니다.

2-1. 강화학습

강화학습은 행동심리학에서 영감을 받은 학습법입니다.

주어진 환경 안에서 에이전트가 현재의 상태를 인식하고

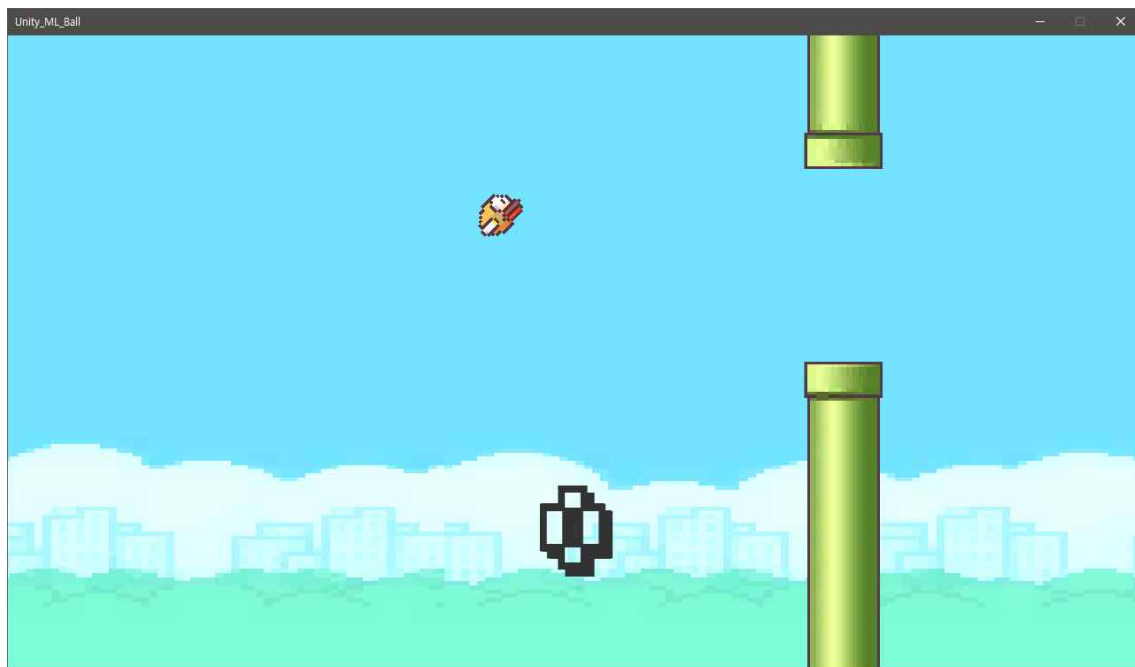
선택 가능한 행동 중 최대한의 보상을 선택하도록 하는 학습법입니다.

2-2. Flappy Bird 게임 해보기

먼저 UnityML 폴더에 있는 Flappy Bird 실행파일을 실행합니다.

아래와 같이 게임이 실행됩니다.

먼저 아무것도 알려드리지 않겠습니다. 즐겨보세요!



아무것도 알지 않은 상태에서 플레이 했음에도 불구하고 어떻게 진행하는지 자연스럽게 알 수 있습니다.

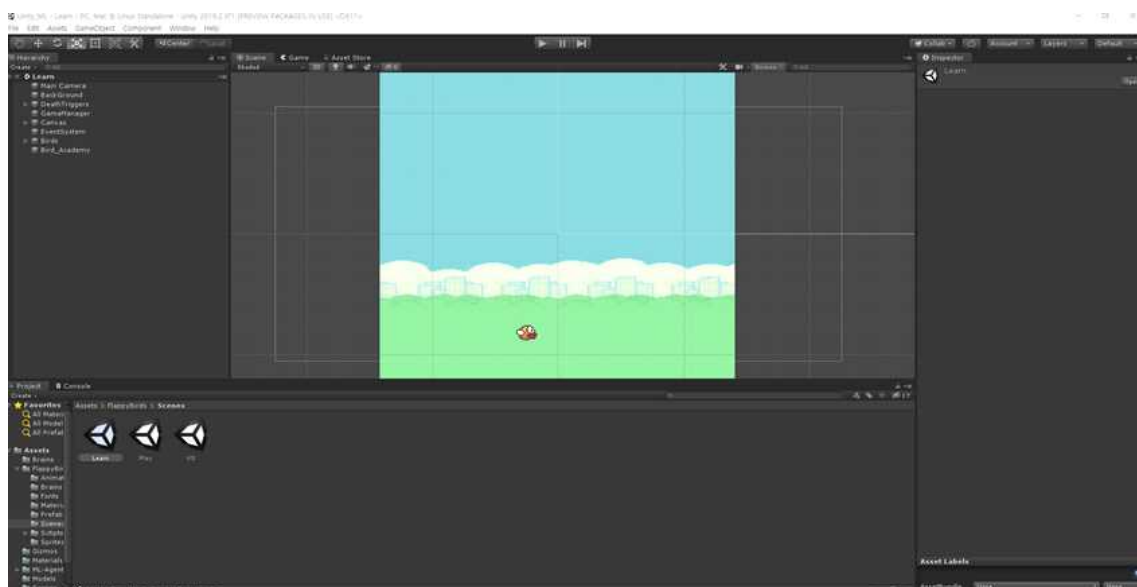
주어진 환경인 게임 안에서 에이전트인 사람은 선택 가능한 행동 중 최대한의 보상을 받는 행동인 점수를 올리는 것을 자연스럽게 하는 것입니다. 컴퓨터에게도 같은 방식으로 학습을 시켜보도록 하겠습니다.

2-3. Flappy Bird 게임 시켜보기

Unity Hub를 실행하여 UnityML 폴더 내부에 있는 Unity_ML선택해 실행시켜줍니다.



방금 플레이한 게임인 Flappy Bird를 만든 툴, Unity가 열리게 됩니다.



Unity 창은 그대로 두고 UnityML 폴더로 이동해 Anaconda.bat 파일을 열어줍니다.

명령 프롬프트 창이 뜨면 Anaconda.txt 파일을 열어 내용을 한줄 씩 입력 해줍니다.

아래의 내용을 복사해도 됩니다.

```
activate mlagents
```

```
mlagents-learn --train --run-id= "FlappyBird" trainer_config.yaml
```



```
C:\WINDOWS\system32\cmd.exe - mlagents-learn --train --run-id=FlappyBird trainer_config.yaml
mlagents) C:\Users\SongD_Home\Desktop\UnityML\Unity_ML>mlagents-learn --train --run-id=FlappyBird trainer_config.yaml

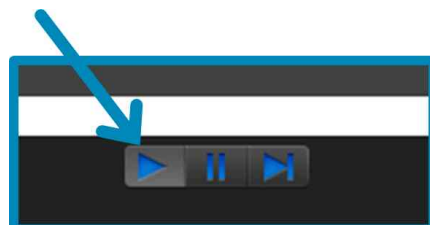
      <-----> unity

INFO:mlagents.trainers:({'--base-port': '5005',
  '--curriculum': 'None',
  '--debug': False,
  '--docker-target-name': 'None',
  '--env': 'None',
  '--help': False,
  '--keep-checkpoints': '5',
  '--lesson': '0',
  '--load': False,
  '--multi-gpu': False,
  '--non-graphics': False,
  '--num-envs': '1',
  '--num-runs': '1',
  '--run-id': 'FlappyBird',
  '--sampler': 'None',
  '--save-freq': '50000',
  '--seed': '-1',
  '--slow': False,
  '--train': True,
  '<trainer-config-path>': 'trainer_config.yaml'})
INFO:mlagents.envs:Start training by pressing the Play button in the Unity Editor.
```

명령 프롬프트 창에 위와 같이 Unity 로고가 나온 후

Start training by pressing the Play button in the Unity Editor

문구를 확인했다면 Unity로 돌아가 재생버튼을 눌러 강화학습을 시작합니다.



강화학습이 진행되는 동안 어떻게 학습이 진행되고 있는지 그래프로 확인해 보도록 하겠습니다.

학습 중인 창을 그대로 두고 UnityML 폴더로 이동합니다.

Tensorboard.bat 파일을 열어 명령 프롬프트 창에 Tensorboard.txt 파일의 내용을 입력해줍니다.

아래의 내용을 복사해도 됩니다.

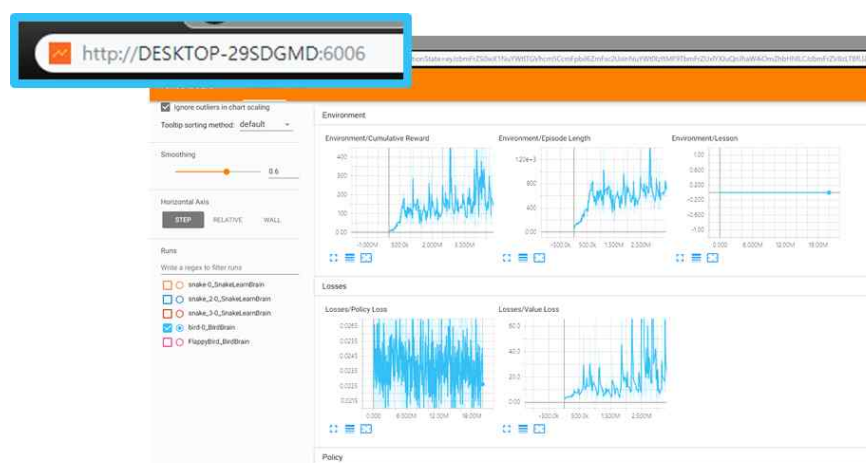
activate mlagents

tensorboard --logdir=summaries

```
C:\WINDOWS\system32\cmd.exe - tensorboard --logdir=summaries
ValueError: A logdir must be specified when db is not specified. Run 'tensorboard --help' for details and examples.
mlagents) C:\Users\Sonad_Home\Desktop\UnityML\UnityML>cd summaries
mlagents) C:\Users\Sonad_Home\Desktop\UnityML\UnityML\summaries>tensorboard
tensorboard: most recent call last:
  File "c:\users\sonad_home\conda\envs\mlagents\lib\runpy.py", line 193, in _run_module_as_main
    "__main__", mod_spec)
  File "c:\users\sonad_home\conda\envs\mlagents\lib\runpy.py", line 85, in _run_code
    exec(code, run_globals)
  File "C:\Users\Sonad_Home\conda\envs\mlagents\Scripts\tensorboard.exe\__main__.py", line 9, in <module>
  File "c:\users\sonad_home\conda\envs\mlagents\lib\site-packages\tensorboard\main.py", line 38, in run_main
    tf.app.run(main)
  File "c:\users\sonad_home\conda\envs\mlagents\lib\site-packages\tensorflow\python\platform\app.py", line 126, in run
    _sys.exit(main(argv))
  File "c:\users\sonad_home\conda\envs\mlagents\lib\site-packages\tensorboard\main.py", line 45, in main
    default.get_assets_zip_provider())
  File "c:\users\sonad_home\conda\envs\mlagents\lib\site-packages\tensorboard\program.py", line 166, in main
    tb = create_tb_app(plugins, assets_zip_provider)
  File "c:\users\sonad_home\conda\envs\mlagents\lib\site-packages\tensorboard\program.py", line 190, in create_tb_app
    raise ValueError('A logdir must be specified when db is not specified.')
ValueError: A logdir must be specified when db is not specified. Run 'tensorboard --help' for details and examples.
mlagents) C:\Users\Sonad_Home\Desktop\2019-09-19_01:28:37_045821_1_1_TensorBoard>
mlagents) C:\Users\Sonad_Home\Desktop>
019-09-19 01:28:37 045821 1 1 TensorBoard 1.7.0 at http://DESKTOP-29SDGMD:6006 (Press CTRL+C to quit)
```

가장 마지막 줄에 있는 주소를 복사 해 인터넷 창에 입력 해 줍니다.

아래와 같은 페이지에서 진행 상황을 확인할 수 있습니다.



UnityML 폴더 내부의 FlappyBird_VS_Computer 실행파일을 통해 위의 학습 과정을 미리 학습시켜 둔 컴퓨터와 대결 해 볼 수 있습니다.

머신러닝의 가장 대표적인 학습법인 지도학습 - 이미지분류 와 비지도학습
- 강화학습에 대해 실습해보았습니다.

정답을 알려주고 학습시키는 학습법인 지도학습과 주어진 환경에서 최대한
의 보상을 얻도록 하는 강화학습에 대해 알 수 있었습니다.

머신러닝을 여행하는 예술가를 위한 안내서 - 실습 끝.
감사합니다.

DADASMIMI
송하윤