

Machine Learning HW3

TAs

ntumlta2018@gmail.com

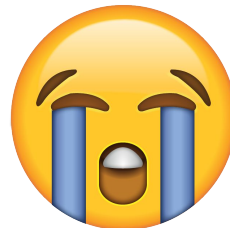
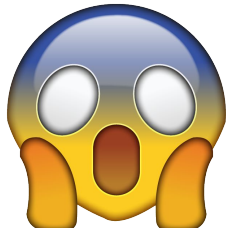
Outline

1. Requirements
2. Task Introduction
3. Data Format
4. Kaggle
5. Rules, Deadline and Policy
6. FAQ

Requirements

1. 請使用**CNN**實作model
2. 不能使用額外data, 不能使用pre-trained model
3. 請附上訓練好的model (及其參數), 若檔案太大請放到github release或dropbox, 並於hw3_test.sh中寫下載的command (請參照以下網站中方法: <http://slides.com/sunprinces/deck-16#/2>)
4. hw3_test.sh要在10分鐘內跑完
5. Available Toolkit Versions:
 - a. Only **Python3.5+**
 - b. 僅可使用 **numpy, pandas, PIL** 以及python standard library
 - c. 可額外使用 **tensorflow1.4.0, keras2.0.8, pytorch0.3.0**
 - d. Report 可以使用 matplotlib, sklearn等library. (但交上來的code不可以import 這兩個, 否則零分)

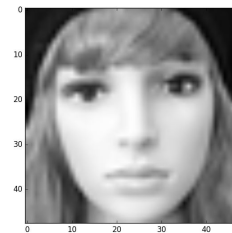
Task - Image Sentiment Classification



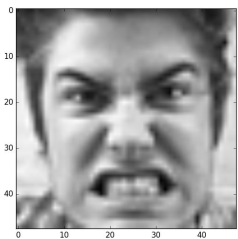
Task - Image Sentiment Classification

本次作業為網路上收集到的人臉表情資料，

經過特殊處理，每張圖片，均是人臉部份佔大部分



6(中立)



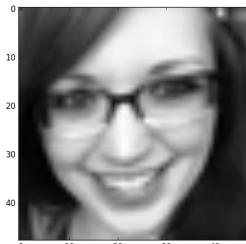
0(生氣)



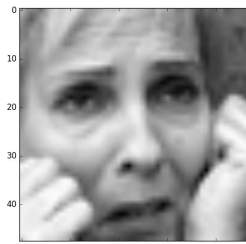
1(厭惡)



2(恐懼)



3(高興)



4(難過)



5(驚訝)

training data: about 28000 images

testing data: about 7000 images (a half will be public test)

Data Format

48

[label], [48*48個灰階強度(intensity)值] (0為黑, 255為白)

<hint> use numpy.reshape function

toy.csv

buffers

```
1 0,170 118 101 88 88 75 78 82 66 74 68 59 63 64 65 90 89 73 80 80 85 88 95 117 132 146 139 152 164 1
81 182 194 195 162 142 147 151 148 158 127 113 112 97 117 117 112 131 124 168 125 108 94 95 77 94 8
7 77 85 69 63 72 74 77 82 106 102 93 98 95 102 98 119 145 168 165 160 168 178 183 178 195 183 151 1
39 145 144 154 143 115 106 122 107 100 111 124 133 173 134 113 101 94 85 104 89 88 91 69 66 75 85 9
0 98 94 130 122 116 107 114 118 117 150 170 184 173 170 176 183 174 180 191 170 146 140 140 143 154
118 119 119 103 91 103 114 128 176 145 115 106 90 95 107 90 99 91 72 72 75 87 101 116 105 118 148
138 122 115 134 127 139 164 183 187 174 173 179 180 166 178 178 158 143 140 137 148 143 117 99 107
104 91 104 116 178 151 119 111 90 105 105 95 109 87 79 85 86 83 102 129 125 121 147 156 148 121 129
147 140 152 170 186 184 169 172 176 171 163 177 169 156 141 134 152 152 101 103 99 109 101 96 109
181 154 125 109 94 111 102 102 108 79 85 91 93 87 87 122 146 128 147 160 164 149 123 143 149 150 15
9 172 184 182 168 173 174 167 166 174 170 152 150 145 147 116 97 100 99 107 97 102 183 153 132 108
101 110 100 113 99 79 91 97 98 98 95 100 144 146 139 170 166 168 144 132 149 149 156 164 173 184 17
9 167 173 171 161 167 171 174 154 130 139 139 90 99 97 100 101 102 183 152 138 108 105 108 102 119
89 84 94 100 103 104 102 99 121 157 147 159 181 167 172 148 139 146 145 150 163 175 185 176 166 172
168 161 176 177 155 135 126 146 110 93 97 91 102 101 185 159 139 110 108 105 109 117 84 92 100 105
109 109 111 107 108 140 162 151 172 182 170 174 149 146 148 143 150 167 180 186 173 164 168 168 16
8 167 167 142 124 131 138 88 96 89 94 104 178 163 136 109 107 102 119 107 85 95 102 107 113 117 121
117 113 118 153 165 159 178 182 169 169 163 159 158 156 159 170 179 187 177 164 166 166 164 167 15
6 122 117 142 116 87 92 87 102 174 155 135 99 107 115 132 105 102 101 101 109 111 117 123 124 116 1
18 126 157 169 164 179 186 171 170 171 169 169 158 161 170 180 180 171 169 167 159 162 139 103
111 138 100 88 77 93 175 154 126 100 113 113 118 91 100 96 89 95 110 114 114 136 147 119 125 131 15
```

NORMAL

tony

10%

L

1/10

:

1

Kaggle

1. kaggle_url: <https://www.kaggle.com/t/b98f465091ab439d90e00665b4250064>
 2. 請至kaggle創帳號登入，需綁定NTU信箱。
 3. 個人進行，不需組隊。
 4. 隊名:學號_任意名稱(ex. b03902000_大金一級棒)，旁聽同學請**避免**學號開頭。
 5. 每日上傳上限**5**次。
 6. test set的資料將被分為兩份，一半為public，另一半為private。
 7. 最後的計分排名將以**2**筆自行選擇的結果，測試在private set上的準確率為準。
- ★ kaggle名稱錯誤者將不會得到任何kaggle上分數。
 - ★ 本次作業為期三週，strong baseline將在第二週結束時公布

Kaggle submission format

請預測test set中七千多筆資料並將結果上傳Kaggle

1. 上傳格式為csv。
2. 第一行必須為id,label, 第二行開始為預測結果。
3. 每行分別為id以及預測的label, 請以逗號分隔。
4. Evaluation: Accuracy

1	id, label
2	0,0
3	1,0
4	2,0
5	3,0
6	4,0
7	5,0
8	6,0
9	7,0
10	8,0
11	9,0
12	10,0
13	11,0
14	12,0
15	13,0
16	14,0
17	15,0
18	16,0
19	17,0
20	18,0
21	19,0

Deadline

- Kaggle: 2018/04/18 23:59:59 (GMT+8)
- report, github: 2018/04/19 23:59:59 (GMT+8)

助教會在deadline一到就clone所有程式, 並且**不再重新clone任何檔案**

Policy

github上ML2018SPRING/hw3/裡面請至少包含：

1. Report.pdf
2. hw3_train.sh
3. hw3_test.sh
4. your python files
5. model參數 (Make sure it can be downloaded by your script.)

(*請將model download到與script相同的位置)

請不要上傳dataset, 請不要上傳dataset, 請不要上傳dataset

Policy

1. 以下的路徑, 助教在跑的時候會另外指定, 請保留可更改的彈性, 不要寫死
2. Script usage:

`bash hw3_train.sh <training data>`

training data: train.csv的路徑

`bash hw3_test.sh <testing data> <prediction file> <mode>`

testing data: test.csv的路徑

prediction file: 結果的csv檔路徑

(除非有狀況, 不然原則上助教只會跑testing, 不會跑training, 因次請用讀取model參數的方式進行predict。)

mode: public或private

Score - Kaggle Rank

- (0.8%) 超過public leaderboard的simple baseline分數
- (0.8%) 超過public leaderboard的strong baseline分數
- (0.8%) 超過private leaderboard的simple baseline分數
- (0.8%) 超過private leaderboard的strong baseline分數
- (0.8%) 2018/04/11 23:59 (GMT+8)前超過public simple baseline
- (BONUS 1%) kaggle排名前五名(且願意上台跟大家分享的同學)
- 其中, 前五名排名以public平均為準, 屆時助教會公布名單

Score - Other Policy

- script 錯誤, 直接0分。若是格式錯誤, 請在公告時間內找助教修好, 修完此次分數*0.7。
- Kaggle超過deadline直接shut down, 可以繼續上傳但不計入成績。
- Github遲交一天(*0.7), 不足一天以一天計算, 不得遲交超過兩天, 有特殊原因請找助教。
- Github遲交表單: <https://goo.gl/forms/xZePqDmE7MrJv9AB3>

(遲交才必需填寫)

遲交請「先上傳程式」Github再填表單, 助教會根據表單填寫時間當作繳交時間。

Score - Other Policy

- 每個人最多可以上傳兩個model, 一個是可以reproduce public的model, 另一個是可以reproduce private的model。
- 如果覺得一個model夠了, 那p11頁testing指令的第三個參數<mode>就都讀同一個model。
- 如果是上傳兩個model的, 那請依照<mode>參數選擇要讀取的model。
- 由於最多可以上傳兩個model, 所以這次作業reproduce出來的結果必須跟kaggle上的一樣, 如果不一樣, 就是那個<mode>的分數0分, 沒有容許誤差範圍。

請記得隨時存model !!!

Score - Report.pdf

[注意] 這次報告建議頁數為三頁，請盡量精簡內容

- (1%) 請說明你實作的 CNN model, 其模型架構, 訓練參數和準確率為何?
- (1%) 請嘗試 data normalization, data augmentation, 說明實行方法並且說明對準確率有什麼樣的影響?
- (1%) 觀察答錯的圖片中, 哪些 class 彼此間容易用混? [繪出 confusion matrix 分析]
- (1%) 使用 CNN 的確有些好處, 試繪出其 saliency maps, 觀察模型在做 classification 時, 是 focus 在圖片的哪些部份?
- (1%) 承(4) 利用上課所提到的 gradient ascent 方法, 觀察特定層的 filter 最容易被哪種圖片 activate 與觀察 filter 的 output。
- 詳情可以參考作業網頁。

Report template:

https://docs.google.com/document/d/1SvPbF7OisOj_MGTNi8bmvg4bLxaN5Z4xJ5v3xDmjmFE/edit

小老師制度(手把手教學)

- 在4/11以前超過simple baseline並願意在4/12在上課時間教導同學撰寫作業三程式，請填寫以下表單：
- 4/12 將公布小老師名單在作業網頁，人數太多將以符合以下標準的同學為主：
 1. 沒有當過小老師
 2. Kaggle Public Leaderboard成績排名較高（但請不要因此想overfit public set)
- 小老師當次成績+1%

FAQ

- 若有其他問題，請po在FB社團裡或寄信至助教信箱，**請勿直接私訊助教。**
- 助教信箱：ntumlta2018@gmail.com

Link

1. 作業網頁: <https://ntumlta2018.github.io/ml-web-hw3/>
2. Kaggle : <https://www.kaggle.com/t/b98f465091ab439d90e00665b4250064>
3. 小老師表單: <https://goo.gl/forms/XpUU5mltlsc90ZW42>
4. TA_hours:
<https://docs.google.com/presentation/d/1ldIMFtmpY25ptCZuBr2UVmZYx4RfgIBOhhesG4VNnEU/edit?usp=sharing>
5. 遲交表單: <https://goo.gl/forms/xZePqDmE7MrJv9AB3>

Tensorflow



TensorFlow

<https://www.tensorflow.org/>



How to Install Tensorflow

- Direct step:
 - install tensorflow

```
> sudo pip install tensorflow
```

- Install in virtual environment:
 - install pyenv
 - install anaconda-2.x.x
 - install tensorflow

```
> pip install tensorflow
```

1. install pyenv

- Mac OSX Homebrew

```
> brew update  
> brew install pyenv
```

- From Github

(for ubuntu user, please replace .bash_profile with .bashrc)

```
> cd ~  
> git clone https://github.com/yyuu/pyenv.git ~/.pyenv  
> echo 'export PYENV_ROOT="$HOME/.pyenv"' >> ~/.bash_profile  
> echo 'export PATH="$PYENV_ROOT/bin:$PATH"' >> ~/.bash_profile  
> echo 'eval "$(pyenv init -)"' >> ~/.bash_profile  
> source ~/.bash_profile
```

install anaconda-2.x.x and Tensorflow

- install anaconda-2.4.0

```
> pyenv install anaconda-2.4.0
```

- switch to anaconda 2.4.0

```
> pyenv global anaconda-2.4.0
```

- Install tensorflow

```
> pip install tensorflow
```

- PyTorch Web: <http://pytorch.org/>

Get Started.

Select your preferences, then run the PyTorch install command.

Please ensure that you are on the latest pip and numpy packages.
Anaconda is our recommended package manager

OS	Linux	OSX	
Package Manager	conda	pip	Source
Python	2.7	3.5	3.6
CUDA	7.5	8.0	None

Run this command:

```
conda install pytorch torchvision cuda80 -c soumith
```

Keras

- **Install Keras from PyPI (recommended):**

```
sudo pip install keras
```

If you are using a virtualenv, you may want to avoid using sudo:

```
pip install keras
```