Python介紹

數據科學的時代

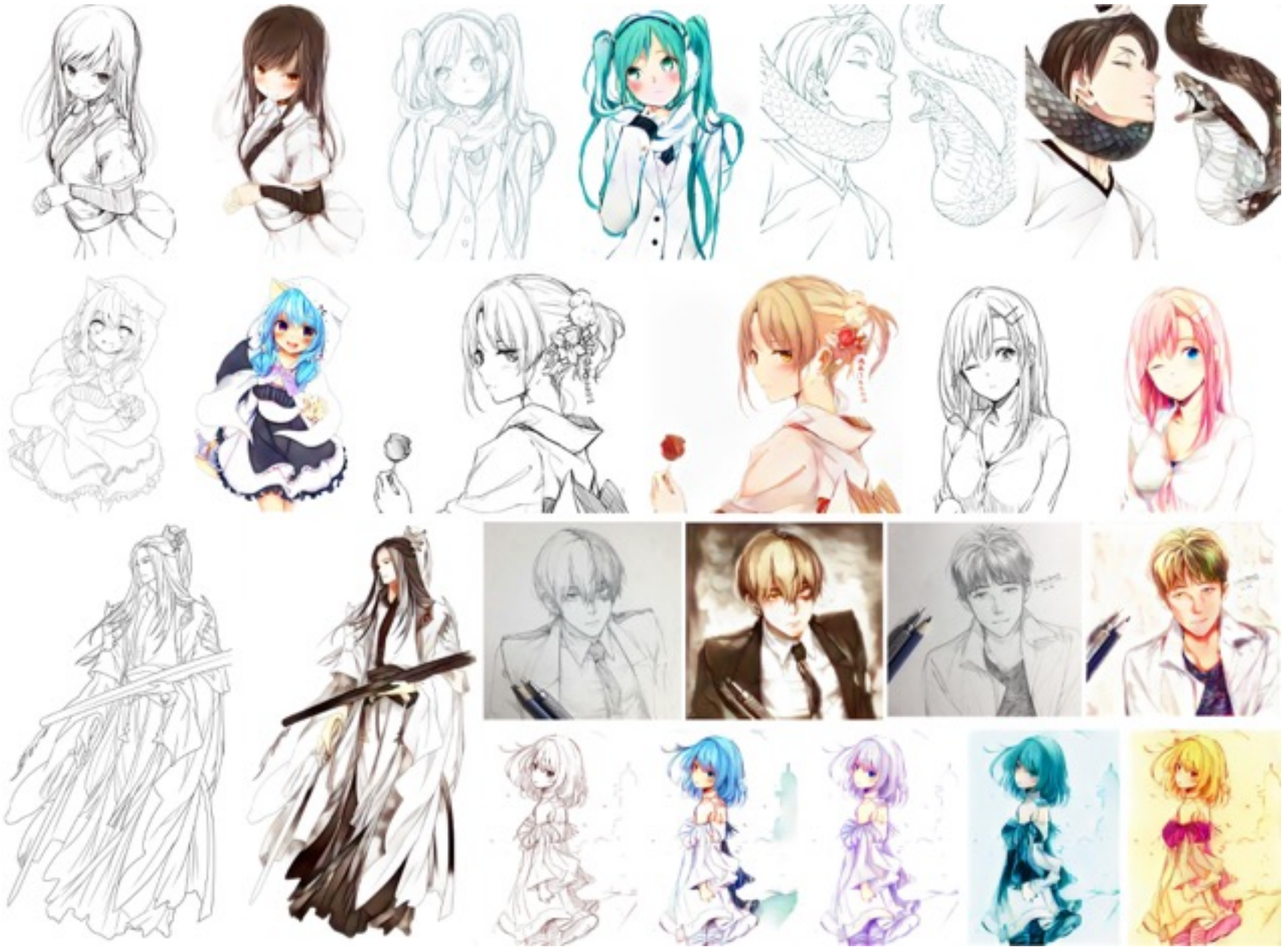
- 我們常常聽見這句話：『人人都要懂數據科學！』，這句話到底是為什麼呢？主要是因為不管什麼職業，數據的量級都因為『網路』和『電腦』的出現以及進步大量的提升！

數據科學到底是什麼？

- 我喜歡把『數據科學』分成兩部分來說
    - 數據收集：透過『網路』和『社群網路』的發達，你可以輕易的收集到超大量的數據
    - 數據分析：超大量的數據最適合的『分析方式』就是『統計』和『機率』了，換個詞來說其實就是『A.I.』

現在數據科學家到底在做什麼?(1)

- https://github.com/lllyasviel/style2paints
- 線稿自動上色工具



現在數據科學家到底在做什麼?(2)

- https://github.com/yunjey/StarGAN

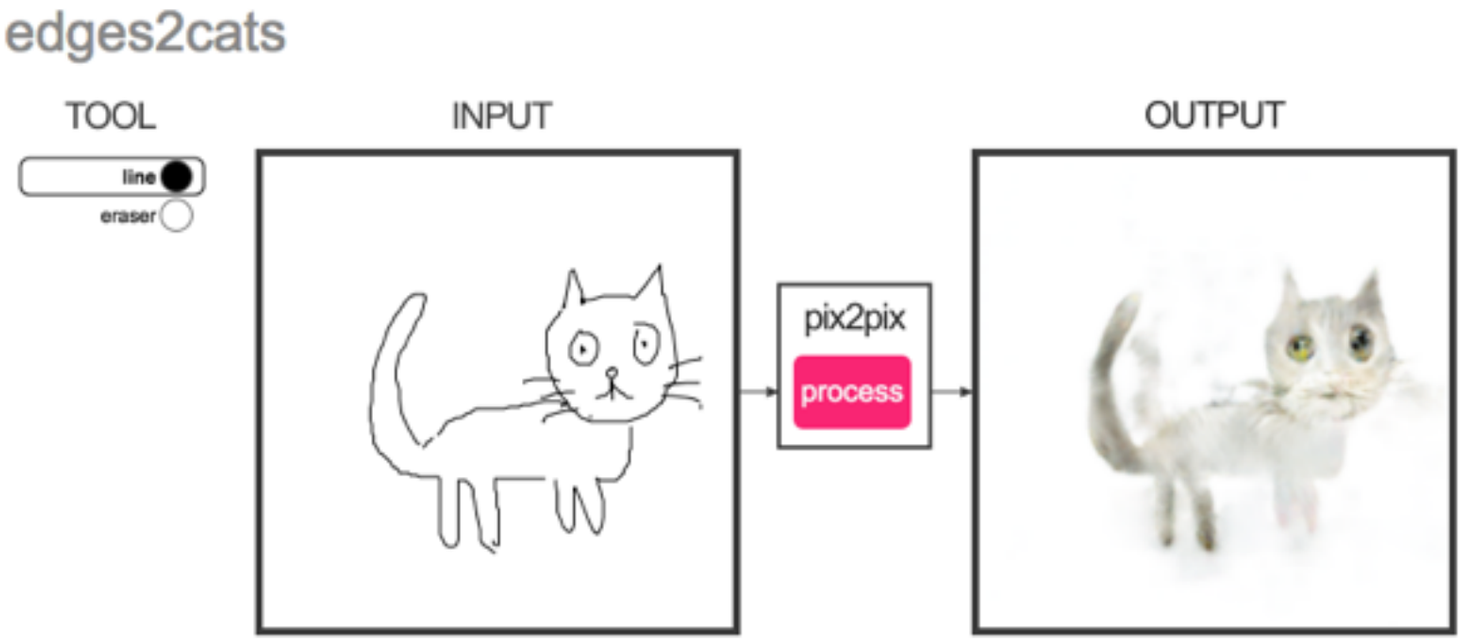
- 自動調整表情

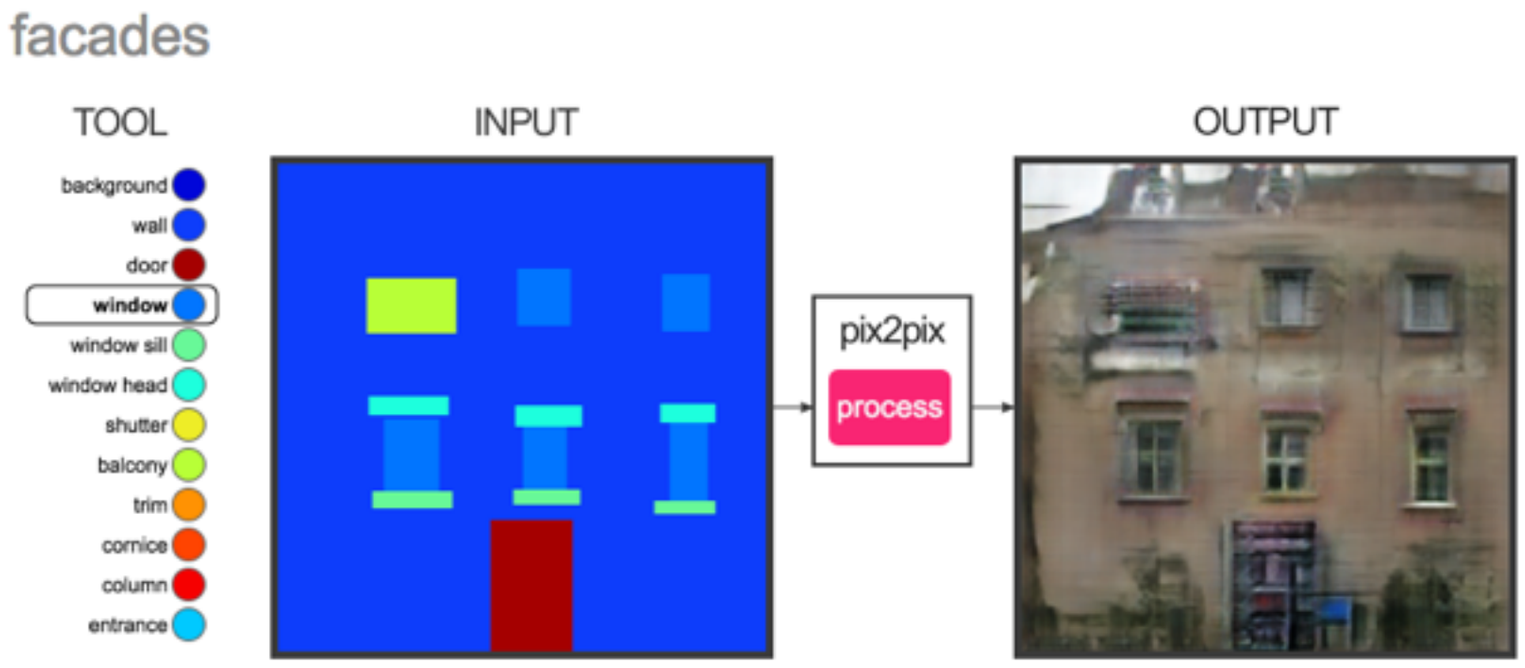
現在數據科學家到底在做什麼?(3)

- https://github.com/junyanz/CycleGAN
- 自動風格轉換

你也可以玩玩看！

- https://affinelayer.com/pixsrv/



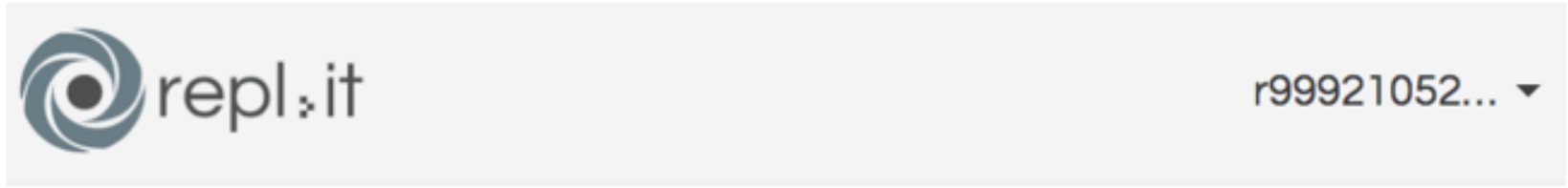


簡單程式學習

快速的練習一些基本的語法

線上程式編寫環境

- https://repl.it
- 不用複雜的環境架設，隨插即用



中文語言分析

使用TF-IDF方法

自然語言處理

- 自然語言處理(Natural Language Processing)可以幫助你做到
    - 分析文章的主題
    - 分析文章內容和文章所得分數的關聯
    - 分析作者的寫作習慣
    - …等等(不勝枚舉)

自然語言處理大步驟

- Step1. 得到文章
- Step2. 將文章內容切成一個一個字詞, 並且記錄字詞出現次數
- Step3. 透過tf-idf方法來衡量每個字詞的重要性

切割字詞

- 假設一個句子: “我相信愛情”
    - 錯誤切割方法: ”我”, “相”, “信”, “愛”, “情”
        - 每一個切割都是無意義的存在
    - 正確切割方法: “我”, “相信”, “愛情”
        - 一個切割是一個對於人類有意義的單字
- 英文會較好切割, 因為擁有空白

tf-idf

- 假設我們現在有了個字詞組, 如何衡量字詞的重要性呢
    - tf: (某字詞出現次數)/(所有字詞總和出現次數)
        - 計算某字詞在這文章出現的次數(次數高 = 重要性高)
    - idf: (文章庫總文章數)/(某詞出現的文章數)
        - idf是透過預先讀完文章庫的文章來計算出來(出現在越多文章 = 重要性低)
        - 像”我”這個字, 幾乎在每一篇文章都出現, 他的idf就是1, 但”愛情”這個字, 在少數文章才出現, idf遠大於1
- 一個字詞 tf * idf越高, 越可以代表這文章

Demo

來分析ptt的文章吧

目標

- 隨意找一篇PTT的文章: https://www.ptt.cc/bbs/Boy-Girl/M.1495546810.A.7B0.html
- 在repl.it上開一個txt

來寫程式吧

```python
from jieba.analyse import extract_tags

# 如果你存檔的時候沒刻意選utf-8, 就不用特別寫encoding
# 因為不寫就會用預設編碼開啟, windows使用ANSI, mac使用utf-8
f = open("article.txt", "r", encoding = "utf-8")
article = f.read()
# 第二個參數代表你想要"幾"個關鍵字
print("這篇文章五個最重要的字:", extract_tags(article, 5))
```

```
Building prefix dict from the default dictionary ...
Dumping model to file cache /var/folders/wv/6kw6f9sx5mqdrjl6820mgzfm0000gn/T/jieba.cache
Loading model cost 1.145 seconds.
Prefix dict has been built succesfully.
這篇文章五個最重要的字: ['愛情', '我們', '男友', '相信', '因為']
```

結論

- 自然語言分析可以用在非常多的方面, 只要是人類打的文章, 說的話, 你就可以藉由分析來得出要點, 再經由這些要點歸納結論或者是預測!

英文語言分析

使用RAKE

RAKE

- 利用停用詞和標點將短語標示出
- 每個詞的重要性是由 多少不同種詞與這個詞相鄰決定

```python
from rake_nltk import Rake
import nltk

nltk.download('stopwords')
nltk.download('punkt')

f = open('a.txt')
article = f.read()
f.close()

r = Rake()
r.extract_keywords_from_text(article)
print(r.get_ranked_phrases())
```

```
['others may permit access without enforcing access control',
''wiki ][ note 1 ])', 'editing rights may permit changing', 'top
ten since 2007', 'could possibly work ".', 'online encyclopedia
project wikipedia', 'users collaboratively modify content', 'run
using wiki software', 'permit control', 'simplest online database',
'widely viewed sites', 'knowledge management resources', 'hawaiian
word meaning', 'including blog software', 'content management
system', 'wiki (/ 'wɪki', 'little implicit structure', 'first wiki
software', 'simplified markup language', 'including wikis
functioning', 'five million articles', 'bug tracking systems',
'access );', 'different wiki engines', 'whereas others', 'rules
may', 'created without', 'written using', 'quick ".', 'wiki
engines', 'organize content', 'different functions', 'wiki engine',
'typical wiki', 'single wiki', 'popular wiki', 'structure
directly', 'language wikipedia', 'allowing structure', 'webm
interview', 'web browser', 'ward cunningham', 'sound listen',
'september 2016', 'removing material', 'otherwise known',
'originally described', 'open source', 'often edited', 'notetaking
tools', 'emerge according', 'defined owner', 'community websites',
'text editor', 'largest collection', 'based website', 'software',
'users', 'content', 'wiki', 'wikipedia', 'wikis', 'systems',
'language', 'articles', 'website', 'text', 'collection', 'world',
'wikiwikiweb', 'wik', 'use', 'type', 'thousands', 'tens',
'standalone', 'rich', 'rather', 'ranked', 'public', 'proprietary',
'pronounced', 'private', 'part', 'one', 'needs', 'levels',
'leader', 'kind', 'inventor', 'intranets', 'imposed', 'hundreds',
'help', 'file', 'far', 'example', 'english', 'ee', 'dozens',
'differs', 'developer', 'adding']
```

https://repl.it/@Elwing/1101