HW2 – Regular Expression with Python

邵鏡軒 F74051035

程式目的與說明

1. 將 arXiv 的特定作者相關資訊抓取出來
2. 發表年份使用 originally announced 欄位
3. 若作者不在 Author:欄位，則不列入計算
    a. 出現的是作者的其他名字，譬如說 Ian Goodfellow 與 Ian J. Goodfellow 雖然是同一個人，但是在這裡算不同人
    b. 作者出現在 et al. (88 additional authors not shown)裡面也不列入計算

輸出

1. 將納入計算的 result 的 title 印出
2. 將 co-author 列出，包括作者自己
3. 將每一個年份的發表數列用 bar chart 畫出
    顯示方式 $ eog bar_chart.png

程式碼解說(僅挑重點部分解說)

```
# 先匯入三個需要用到的 library
import urllib.request
import re
import matplotlib.pyplot as plt

# 兩個 RE 形式，其中 find_list 為一整個 result(包括 author, title, date 等等)
find_list = 'arxiv-result[\s\S]*?</li>'
find_author = '>[\s\S]*?</a>'
```

```
# 讀入作者名字，並把空格換成” +” ，以便搜尋
author = input()
search_author=author.replace(' ','+')

# 1. 當遇到"Sorry, your query for"時，搜尋完畢，error 變成非空字串
# 2. 每次搜尋 50 個結果
while error == []:
        url = "https://arxiv.org/search/?searchtype=author&query=" + search_author \
                + "&abstracts=hide&size=50&order=-announced_date_first&start=" +
str(start)
        content = urllib.request.urlopen(url)
        html_str = content.read().decode('utf-8')
        find_warning = "Sorry, your query for"
        error = re.findall(find_warning, html_str)

# 分成三個狀況: (1)沒有搜尋到結果 (2)搜尋完畢 (3)紀錄這一頁搜尋的結果，並且 start+=50，準備進到下一頁
        # stop crawling
        if error != [] and coauthor_list == []:
                print("no result")
        elif error != [] and coauthor_list != []:
                print("[ end of title list ]")

                # print coauthors
                print("\n[ Co-author list ]")
                coauthor_list.sort()

# 每個 co-author 先計算出現幾次，然後刪掉，最後印出
                while coauthor_list != []:
                        count = coauthor_list.count(coauthor_list[0])
                        print(coauthor_list[0] + ": %d times" %(count))
                        del coauthor_list[0:count]

                print("[ end of coauthor list ]")

# 把每個年份有多少結果印出來
                # draw the bar chart
```

```
                max_year=max(year_ans)
                max_year=int(max_year)
                min_year=min(year_ans)
                min_year=int(min_year)

                for i in range(min_year, max_year+1):
                        count=year_ans.count(str(i))

                        # in order to set the y-axis
                        if count > max_count:
                                max_count = count
                        y.append(count)
                        x.append(str(i))
                # set the y-axis
                plt.yticks(range(0,max_count+1,2))
                plt.bar(x,y)
# 將圖片結果儲存在 bar_chart.png
                plt.savefig("bar_chart.png")
                # plt.show()

# 先把每個搜尋的 result 抓出來，再判斷作者欄裡面有沒有我們搜尋的作者，
        # 如果有，接著抓出來 title 跟年份
        # 如果沒有，則跳過
        else:
                #find the result lists
                list_result = re.findall(find_list, html_str)
                for l_r in list_result:
                        # find the coauthors
                        temp_author_list.clear()
                        authors = l_r.split("Authors:</span>")[1].split("</p>")[0].strip()
                        each_author = re.findall(find_author, authors)
# 先把這一個 result 的 co-author 紀錄在一個 temp_list 裡
                        for a_r in each_author:
                                result = a_r.split(">")[1].split("</a")[0].strip()
                                temp_author_list.append(result)

# 看作者是否有在 co-author 裡，如果存在，則把
```

temp_list，加到 coauthor_list(紀錄所有 results 裡的 co-author)裡

```
                    for temp_author in temp_author_list:
# 英文大小寫都可以接受
                        if author.lower() == temp_author.lower():
                            coauthor_list = coauthor_list + temp_author_list

# 抓出 title
                            title = l_r.split("title is-5
mathjax\">")[1].split("</p>")[0].strip()
                            print(title)

# 抓出年份
                            date = l_r.split("originally
announced</span>")[1].split("</p>")[0].strip()
                            year = re.findall("[0-9]+", date)
                            year_ans.append(year[0])

                            break;
        start=start+50
```

程式執行環境步驟

$ python3 crawler.py > result.txt

Ian Goodfellow

執行結果: result.txt

$ vi result.txt

[Author: Ian Goodfellow]
Imperceptible, Robust, and Targeted Adversarial Examples for Automatic Speech Recognition
A Research Agenda: Dynamic Models to Defend Against Correlated Attacks
On Evaluating Adversarial Robustness
New CleverHans Feature: Better Adversarial Robustness Evaluations with Attack Bundling
Discriminator Rejection Sampling
Local Explanation Methods for Deep Neural Networks Lack Sensitivity to Parameter

Values
Sanity Checks for Saliency Maps
Unrestricted Adversarial Examples
Skill Rating for Generative Models
TensorFuzz: Debugging Neural Networks with Coverage-Guided Fuzzing
Understanding and Improving Interpolation in Autoencoders via an Adversarial Regularizer
Motivating the Rules of the Game for Adversarial Example Research
Adversarial Reprogramming of Neural Networks
Defense Against the Dark Arts: An overview of adversarial example security research and future research directions
Self-Attention Generative Adversarial Networks
Gradient Masking Causes CLEVER to Overestimate Adversarial Perturbation Size
Adversarial Attacks and Defences Competition
Adversarial Logit Pairing
Is Generator Conditioning Causally Related to GAN Performance?
Adversarial Examples that Fool both Computer Vision and Time-Limited Humans
MaskGAN: Better Text Generation via Filling in the_____
Adversarial Spheres
Many Paths to Equilibrium: GANs Do Not Need to Decrease a Divergence At Every Step
On the Protection of Private Information in Machine Learning Systems: Two Recent Approaches
Ensemble Adversarial Training: Attacks and Defenses
The Space of Transferable Adversarial Examples
Adversarial Attacks on Neural Network Policies
NIPS 2016 Tutorial: Generative Adversarial Networks
Adversarial Machine Learning at Scale
Semi-supervised Knowledge Transfer for Deep Learning from Private Training Data
Technical Report on the CleverHans v2.1.0 Adversarial Examples Library
Adversarial examples in the physical world
Deep Learning with Differential Privacy
Improved Techniques for Training GANs
Adversarial Training Methods for Semi-Supervised Text Classification
Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples
Unsupervised Learning for Physical Interaction through Video Prediction
Improving the Robustness of Deep Neural Networks via Stability Training
TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems
Practical Black-Box Attacks against Machine Learning
Adversarial Autoencoders
Net2Net: Accelerating Learning via Knowledge Transfer
Efficient Per-Example Gradient Computations
Intriguing properties of neural networks
Joint Training of Deep Boltzmann Machines
Theano: new features and speed improvements
Large-Scale Feature Learning With Spike-and-Slab Sparse Coding

[end of title list]

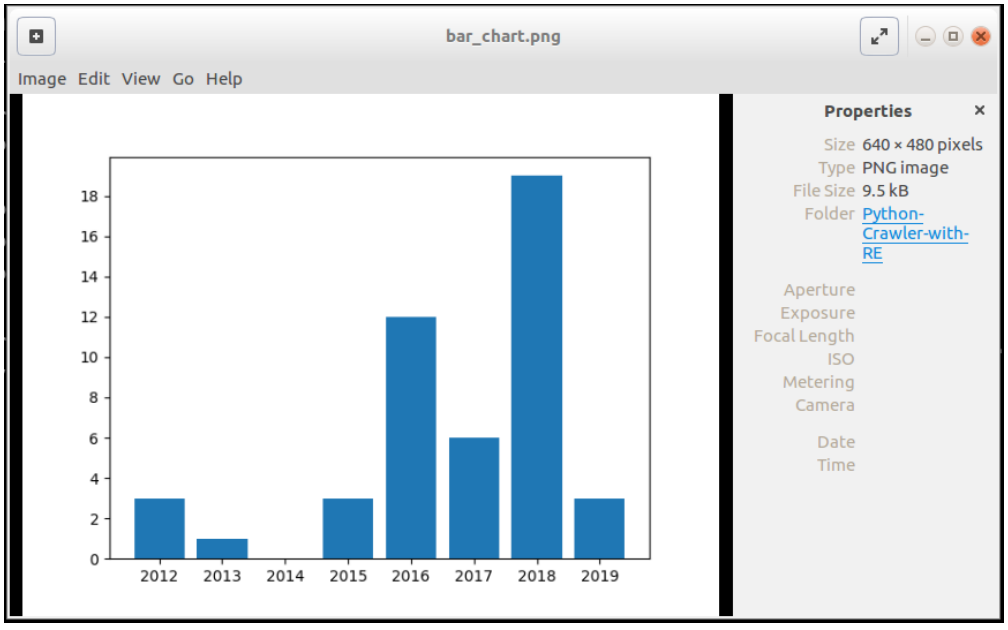
[Co-author list]
Aaron Courville: 2 times
Abhibhav Garg: 1 times
Alan Yuille: 1 times
Alec Radford: 1 times
Aleksander Madry: 1 times
Alex Kurakin: 1 times
Alexander Matyasko: 1 times
Alexey Kurakin: 7 times
Alireza Makhzani: 1 times
Ananthram Swami: 1 times
Andrew Harp: 1 times
Andrew M. Dai: 3 times
Andy Chu: 1 times
Andy Davis: 1 times
Anish Athalye: 1 times
Arnaud Bergeron: 1 times
Ashish Agarwal: 1 times
Augustus Odena: 5 times
Aurko Roy: 2 times
Balaji Lakshminarayanan: 1 times
Been Kim: 2 times
Brendan Frey: 1 times
Brian Cheung: 1 times
Catherine Olsson: 4 times
Chelsea Finn: 1 times
Chiyuan Zhang: 1 times
Chris Olah: 1 times
Christian Szegedy: 1 times
Christopher Olah: 1 times
Cihang Xie: 2 times
Colin Raffel: 3 times
Craig Citro: 1 times
Dan Boneh: 2 times
Dan Mane: 1 times
David Andersen: 1 times
David Berthelot: 2 times
David Warde-Farley: 1 times
Derek Murray: 1 times
Dimitris Metaxas: 1 times
Dimitris Tsipras: 1 times
Dumitru Erhan: 1 times
Eugene Brevdo: 1 times
Fangzhou Liao: 1 times

Fartash Faghri: 2 times
Florian Tramèr: 2 times
Frédéric Bastien: 1 times
Gamaleldin F. Elsayed: 2 times
Garrison Cottrell: 1 times
Geoffrey Irving: 1 times
George E. Dahl: 1 times
Greg S. Corrado: 1 times
H. Brendan McMahan: 2 times
Han Zhang: 1 times
Harini Kannan: 1 times
Ian Goodfellow: 47 times
Ilya Mironov: 2 times
Ilya Sutskever: 1 times
Jacob Buckman: 1 times
James Bergstra: 1 times
Jascha Sohl-Dickstein: 2 times
Jeffrey Dean: 1 times
Jianyu Wang: 1 times
Joan Bruna: 1 times
Jonas Rauber: 2 times
Jonathan Uesato: 1 times
Jonathon Shlens: 2 times
Josh Levenberg: 1 times
Julius Adebayo: 2 times
Jun Zhu: 1 times
Junjiajia Long: 1 times
Justin Gilmer: 4 times
Karen Hambardzumyan: 1 times
Kunal Talwar: 3 times
Li Zhang: 2 times
Lukasz Kaiser: 1 times
Luke Metz: 1 times
Maithra Raghu: 1 times
Manjunath Kudlur: 1 times
Martin Wattenberg: 1 times
Martín Abadi: 4 times
Matthieu Devin: 1 times
Michael Isard: 1 times
Michael Muelly: 1 times
Mihaela Rosca: 1 times
Ming Liang: 1 times
Moritz Hardt: 1 times
Motoki Abe: 1 times
Navdeep Jaitly: 1 times
Nicholas Carlini: 4 times

Nicolas Bouchard: 1 times
Nicolas Papernot: 10 times
Pascal Lamblin: 1 times
Patrick McDaniel: 4 times
Paul Barham: 1 times
Paul Christiano: 1 times
Paul Hendricks: 1 times
Pieter Abbeel: 1 times
Rafal Jozefowicz: 1 times
Rajat Monga: 1 times
Razvan Pascanu: 1 times
Reuben Feinman: 1 times
Rob Fergus: 1 times
Rujun Long: 1 times
Ryan P. Adams: 1 times
Ryan Sheatsley: 1 times
Samaneh Azadi: 1 times
Samuel S. Schoenholz: 1 times
Samy Bengio: 3 times
Sandy Huang: 1 times
Sangxia Huang: 1 times
Sanjay Ghemawat: 1 times
Seiya Tokui: 1 times
Sergey Levine: 1 times
Shakir Mohamed: 1 times
Sherry Moore: 1 times
Shreya Shankar: 1 times
Somesh Jha: 1 times
Stephan Zheng: 1 times
Surya Bhupatiraju: 1 times
Takeru Miyato: 1 times
Takuya Akiba: 1 times
Thomas Leung: 1 times
Tianqi Chen: 1 times
Tianyu Pang: 1 times
Tim Salimans: 1 times
Tom B. Brown: 2 times
Tom Brown: 2 times
Trevor Darrell: 1 times
Vahid Behzadan: 1 times
Vicki Cheung: 1 times
Wieland Brendel: 1 times
Willi Gierke: 1 times
William Fedus: 2 times
Wojciech Zaremba: 2 times
Xi Chen: 1 times

Xiaolin Hu: 1 times
Yan Duan: 1 times
Yang Song: 1 times
Yangqing Jia: 1 times
Yao Qin: 1 times
Yao Zhao: 1 times
Yash Sharma: 1 times
Yerkebulan Berdibekov: 1 times
Yi-Lin Juang: 1 times
Yinpeng Dong: 2 times
Yoshua Bengio: 3 times
Yuzhe Zhao: 1 times
Z. Berkay Celik: 1 times
Zhi Li: 1 times
Zhifeng Chen: 1 times
Zhishuai Zhang: 2 times
Zhonglin Han: 1 times
Zhou Ren: 1 times
Úlfar Erlingsson: 2 times
[end of coauthor list]

$ eog bar_chart.png



其他程式說明

1. [\s\S]*?</p>的*?
   意思是任意長度的最短匹配(non-greedy)
2. str.split(' ',2)
   意思是以空格為區隔方式，區隔 2 次(變成 3 個)，預設為空白符(包含\n 等)及最大分割數
3. str.strip('0')
   把頭尾的很多 0 去掉，預設為空白符
4. title=r.split("title is-5 mathjax">")[1].split("</p>")[0].strip()
   為取一個區間的辦法，然後再去掉頭尾空格