Welcome and Hello!
I'm David

本雲端筆記本網址:

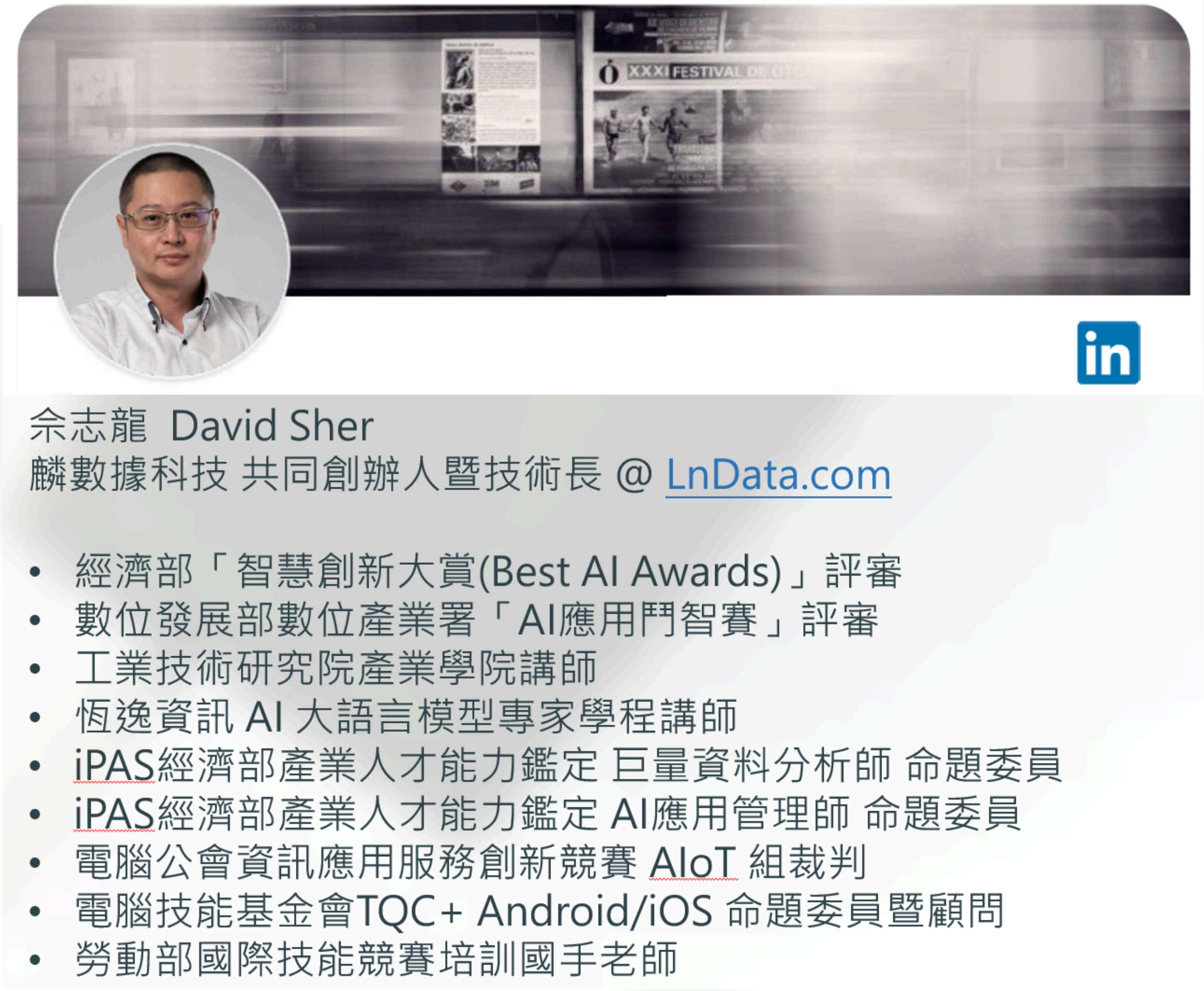
First things first:

1) Install Anaconda. (which is already done by 恆逸)
2) Install Google Chrome browser (http://goo.gl/GC6VbG) (which is already done by 恆逸)
3) Copy C:\!!David\Anaconda\Anaconda3.rar to USB (HOME)
4) Download Environment here: https://bit.ly/2BDyzFs and extract to C:\Programming\
5) Send me an e-mail: DavidLanz@gmail.com with your Gmail account.
6) Accept David's invitation of a shared document on Google Drive.
7) 共享雲端硬碟檔案:
https://drive.google.com/drive/folders/1R7WI4fhd91e07RbV9tFspZcPcVuby1Jt?usp=sharing

I'll continue when you finish.

Notice!
Please do NOT upgrade any Python package or patch during class, thank you.

Contacts:
DavidLanz@gmail.com



佘志龍 David Sher
麟數據科技 共同創辦人暨技術長 @ LnData.com

- 經濟部「智慧創新大賞(Best AI Awards)」評審
- 數位發展部數位產業署「AI應用鬥智賽」評審
- 工業技術研究院產業學院講師
- 恆逸資訊 AI 大語言模型專家學程講師
- iPAS經濟部產業人才能力鑑定 巨量資料分析師 命題委員
- iPAS經濟部產業人才能力鑑定 AI應用管理師 命題委員
- 電腦公會資訊應用服務創新競賽 AIoT 組裁判
- 電腦技能基金會TQC+ Android/iOS 命題委員暨顧問
- 勞動部國際技能競賽培訓國手老師

課程目標
==

【第一階段：初階與基礎應用模組 7 小時】
1. Python爬蟲與資料集構建（0.5小時, Lesson 3, Lesson 4）
2. 自動化標註與文本分類（0.5小時, Lesson 8）
3. NLP 核心模型與演算法實務（1小時, Lesson 9）
4. 新聞/產業資訊分類系統建構（1小時, Lesson 10）
5. 情感分析模型訓練（2小時, Lesson 11）
7. 中文社群聆聽與詞雲分析（2小時, Lesson 12, Lesson Lesson 15, Lesson 19）

【第二階段：RAG + LangChain 實作模組 總計 21 小時】
8. Transformer 與 Transfer Learning（0.5 小時, Lesson 20, Lesson 21）
9. OpenAI 模型 + LangChain RAG（3 小時, Lesson 27, Lesson 28, Lesson 29, Lesson 33）
10. Assistants API + LlamaIndex RAG（2 小時, Lesson 30, Lesson 31, Lesson 35, Lesson 40）
11. GPTs API 串接與 AI Agents 語境優化（2 小時, Lesson 50.1, Lesson 50.2, Lesson 50.3）
12. 開源模型微調：LLaMA / Mistral / Gemma（3 小時, Lesson 24, Lesson 25, Lesson 25.4）
13. RAG 強化知識推理實作（含 CoT 應用）（3 小時, Lesson 46.2, Lesson 51, Lesson 55, Lesson 56）
14. LangChain + 向量資料庫整合，企業知識管理實作（3 小時, Lesson 43, Lesson 45, Lesson 57.1, Lesson 58）
15. LLM GGUF 格式轉換與 CPU 推理部署（1.5 小時）
16. 離線私有化部署 + LangChain RAG 建構流程（3 小時, Lesson 34, Lesson 36）

【Bonus階段：n8n】
17. 安裝 n8n 於 HuggingFace
18. 設計流程、結合表單與更新 GCP 資料

建立 Anaconda 虛擬環境：

==

建立 uuunlp 的虛擬環境 (以管理員身分執行 命令提示字元 **cmd** 指令)

----

1. 於C:\Programming資料夾下，建立一個資料夾名稱為 Conda
2. 以管理員身分開啟 命令提示字元 cmd

cd C:\Programming\
C:\Programming>

3. 下載：requirements.txt

解壓縮放在：C:\Programming\
https://www.dropbox.com/s/ynrhf6ypcyf39na/requirements.rar?dl=0

4. 建立虛擬環境名稱為 uuunlp

conda create -n uuunlp python=3.7

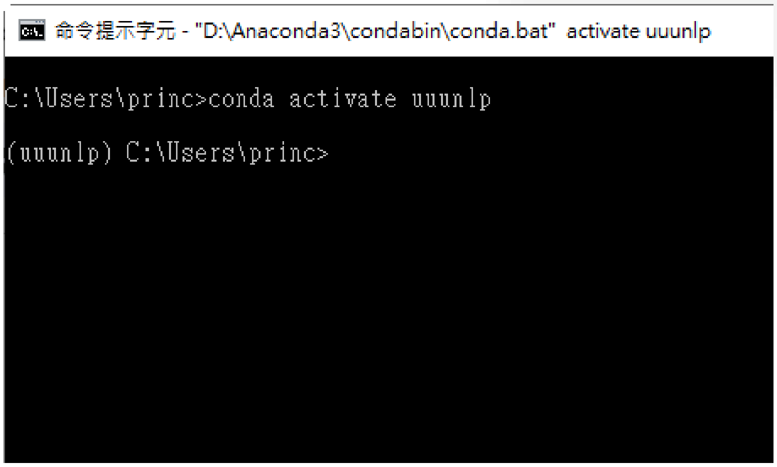
5. 接著於命令提示字元視窗中輸入：

C:\>conda init
(第一次執行 activate uuunlp 需要先執行 conda init 然後關閉 CMD視窗之後重新開啟CMD繼續)

6. 依序執行已下指令進入虛擬環境

關閉命令提示字元視窗後重新以管理員身分開啟命令提示字元
conda activate uuunlp

前方會多了一個虛擬環境的名稱：(uuunlp)


命令提示字元 - "D:\Anaconda3\condabin\conda.bat" activate uuunlp

```
C:\Users\princ>conda activate uuunlp

(uuunlp) C:\Users\princ>
```

以下指令需要再 (uuunlp) C:\Programming> 的 prompt 下執行 (全部複製，貼上至 DOS 視窗)：
–
(uuunlp) C:\Programming> 這個資料夾中應該要有 requirements.txt

複製以下所有 pip install 的指令，貼在 Command Window 中：
===

```
pip install ipykernel
python -m ipykernel install --user --name uuunlp --display-name uuunlp
pip install ipywidgets widgetsnbextension pandas-profiling
pip install notebook==6.5.7
pip install -r requirements.txt
pip install -U ckiptagger
pip install -U transformers
pip install -U ckip-transformers
pip install plac
pip install tabulate
pip install --upgrade nbformat
pip install --upgrade accelerate
pip install protobuf==3.20
pip install h5py==2.10.0
pip install Delorean
```

共享與雲端執行 Jupyter notebook：
============================

```
C:\Anaconda3\Scripts\jupyter.exe notebook --ip="127.0.0.1" --no-browser --port=9487
```

注意：移除虛擬環境的指令 (以下不要在課堂中執行)

```
(uuunlp) C:\>conda deactivate
C:\>conda remove --name uuunlp --all
C:\>jupyter kernelspec uninstall uuunlp
```

最後至 C:\Andconda3\envs\
將 uuunlp 資料夾刪除

列出所有 conda 的虛擬環境：

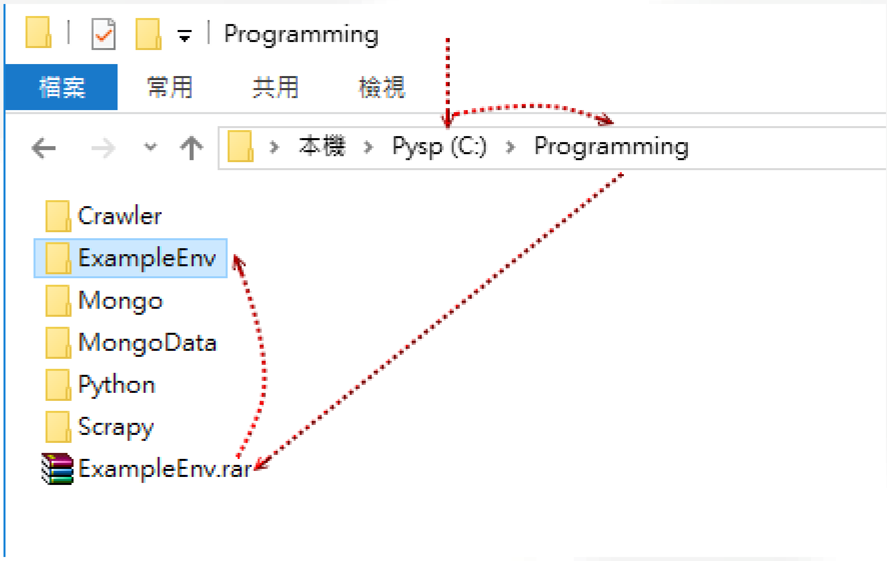
```
C:\>conda env list
```

Environment preparation
==

1. Before we start coding, please download Programming Environment here:
https://www.dropbox.com/s/vbztpjl98kcceuy/ExampleEnv.rar?dl=0

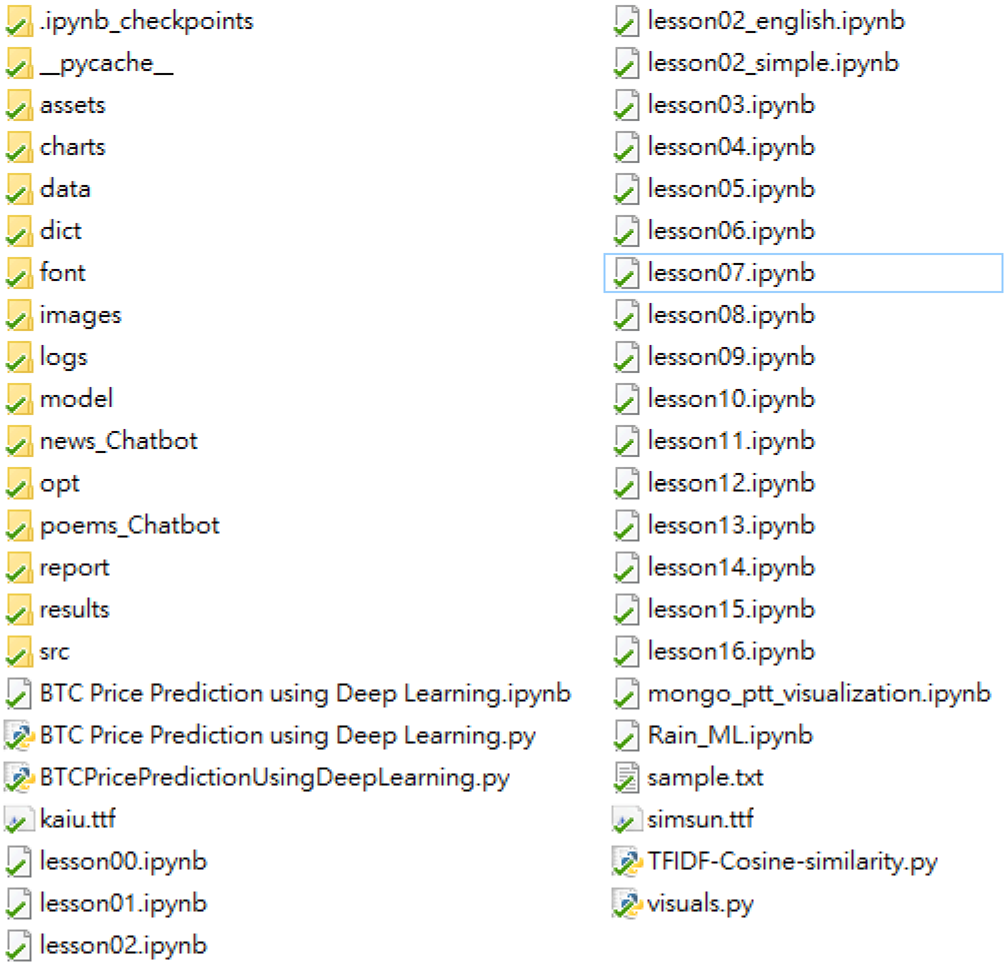
本堂上課環境所有需要的檔案, RAR 解壓縮密碼(小寫英文字母):uuu

2. 將「ExampleEnv.rar」壓縮檔, 解壓縮至 C:\Programming\
資料夾結構如下圖所示:



Crawler
ExampleEnv
Mongo
MongoData
Python
Scrapy
ExampleEnv.rar

3. Download Source code and put them under Environment folder:
(David will provide each sample during the class.)



.ipynb_checkpoints
__pycache__
assets
charts
data
dict
font
images
logs
model
news_Chatbot
opt
poems_Chatbot
report
results
src
BTC Price Prediction using Deep Learning.ipynb
BTC Price Prediction using Deep Learning.py
BTCPricePredictionUsingDeepLearning.py
kaiu.ttf
lesson00.ipynb
lesson01.ipynb
lesson02.ipynb
lesson02_english.ipynb
lesson02_simple.ipynb
lesson03.ipynb
lesson04.ipynb
lesson05.ipynb
lesson06.ipynb
lesson07.ipynb
lesson08.ipynb
lesson09.ipynb
lesson10.ipynb
lesson11.ipynb
lesson12.ipynb
lesson13.ipynb
lesson14.ipynb
lesson15.ipynb
lesson16.ipynb
mongo_ptt_visualization.ipynb
Rain_ML.ipynb
sample.txt
simsun.ttf
TFIDF-Cosine-similarity.py
visuals.py

Lesson 3 - Crawler & Fetch training data

Table of Contents

Source code: https://www.dropbox.com/s/z7lkhxb3xpq8qwn/lesson03.rar?dl=0

輸入指令 jupyter notebook 執行 notebook

```
cd Programming\ExampleEnv>conda activate uuunlp
(uuunlp) C:\Programming\ExampleEnv>jupyter notebook
```

<1> 下載對應自己 Chrome 瀏覽器的版本 Chromedriver
https://googlechromelabs.github.io/chrome-for-testing/#stable
解壓縮至 C:\Programming\ExampleEnv\Assets\

<2> 安裝 Sublime 套件：Install Pretty JSON package in Sublime Text 3
===
0. Sublime Install package：
1. CTRL+SHIFT+P -> Install Package
2. CTRL+SHIFT+P -> Package Control - Install Package
3. JSON -> Pretty JSON
4. Ctrl+ALT+J

Install Pretty JSON package in Sublime Text 4
Sublime Text 4, Preferences > Key Bindings
===

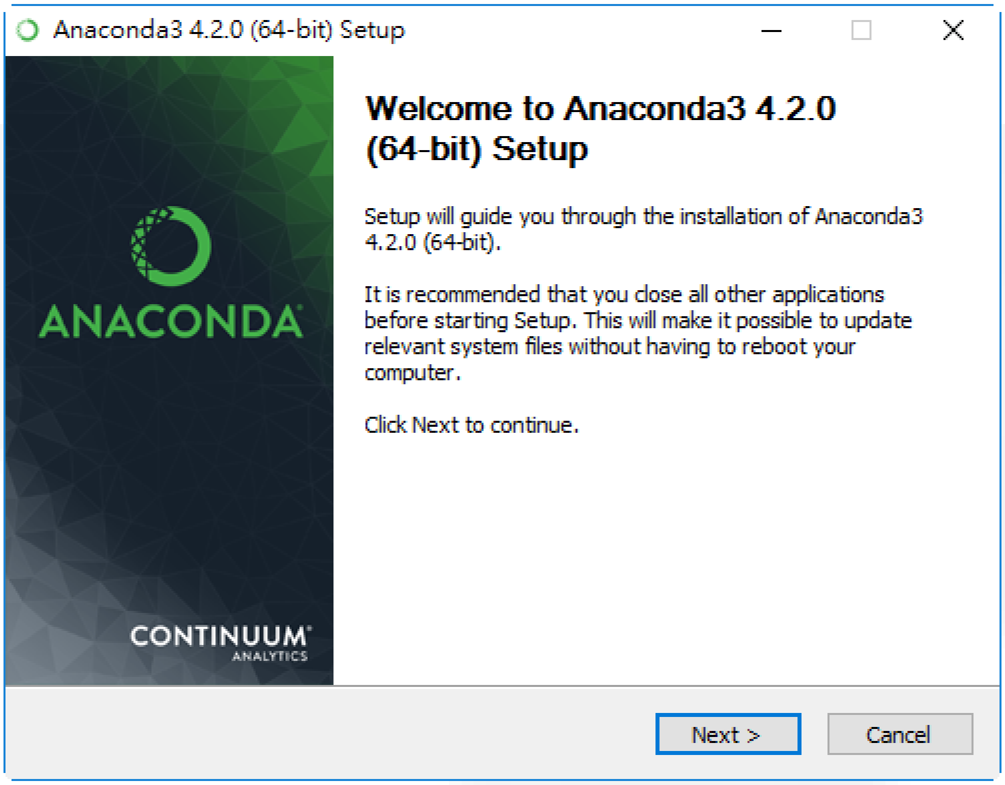
```
[
    {
        "keys": [
            "ctrl+alt+j"
        ],
        "command": "pretty_json"
    },
    {
        "keys": [
            "ctrl+alt+m"
        ],
        "command": "un_pretty_json"
    },
    {
        "keys": ["ctrl+r"],
        "command": "pretty_json_goto_symbol",
        "context": [
            { "key": "selector", "operator": "equal", "operand": "source.json" }
        ]
    }
]
```

```
break;
```

本堂課環境的安裝，下載並安裝 Anaconda

===

下載：https://www.anaconda.com/products/individual



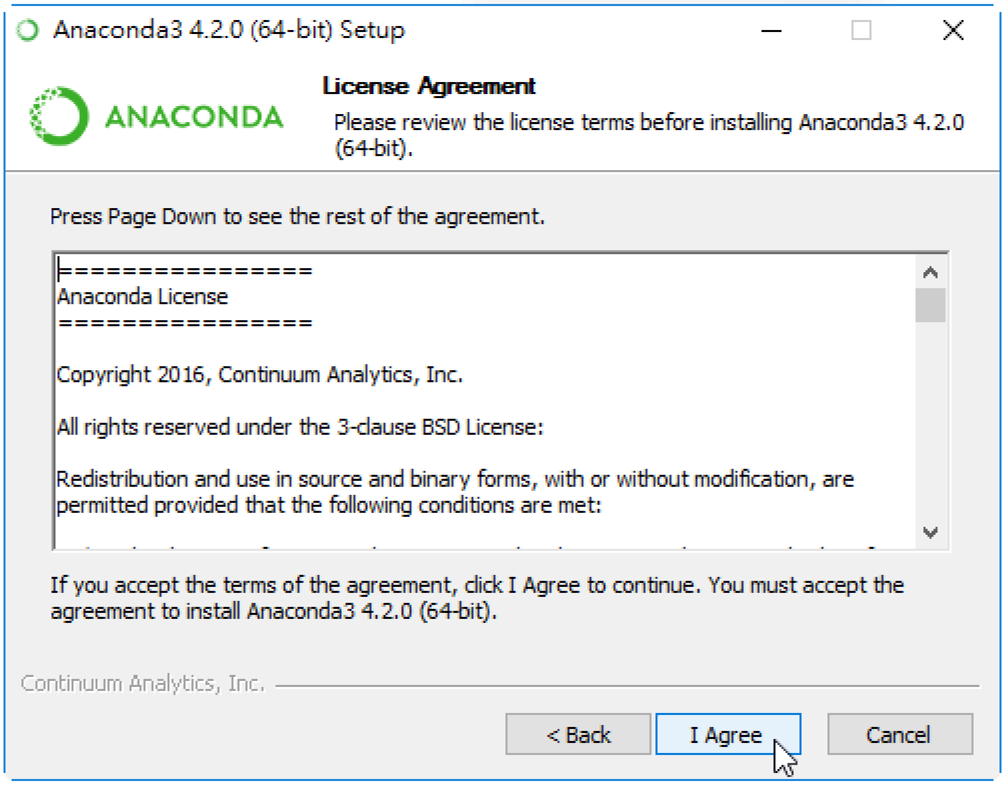
Anaconda3 4.2.0 (64-bit) Setup

Welcome to Anaconda3 4.2.0 (64-bit) Setup

Setup will guide you through the installation of Anaconda3 4.2.0 (64-bit).

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.



Anaconda3 4.2.0 (64-bit) Setup

License Agreement

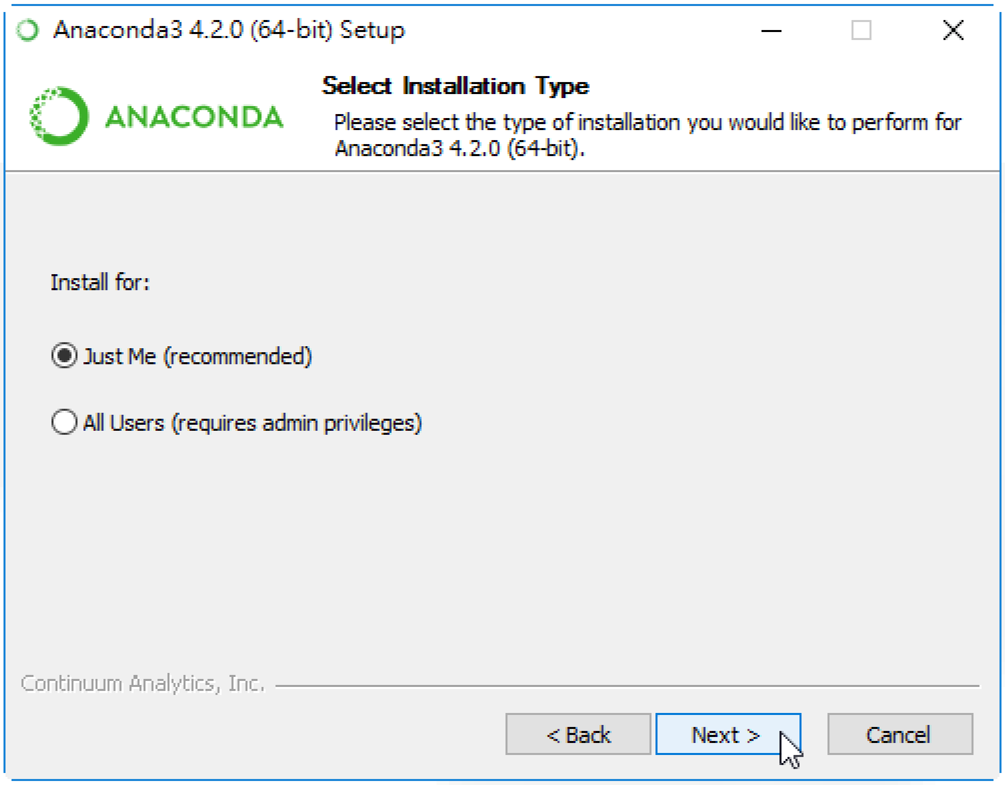
Please review the license terms before installing Anaconda3 4.2.0 (64-bit).

Press Page Down to see the rest of the agreement.

```
================
Anaconda License
================

Copyright 2016, Continuum Analytics, Inc.

All rights reserved under the 3-clause BSD License:

Redistribution and use in source and binary forms, with or without modification, are
permitted provided that the following conditions are met:
```

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Anaconda3 4.2.0 (64-bit).

Continuum Analytics, Inc.



Anaconda3 4.2.0 (64-bit) Setup

Select Installation Type

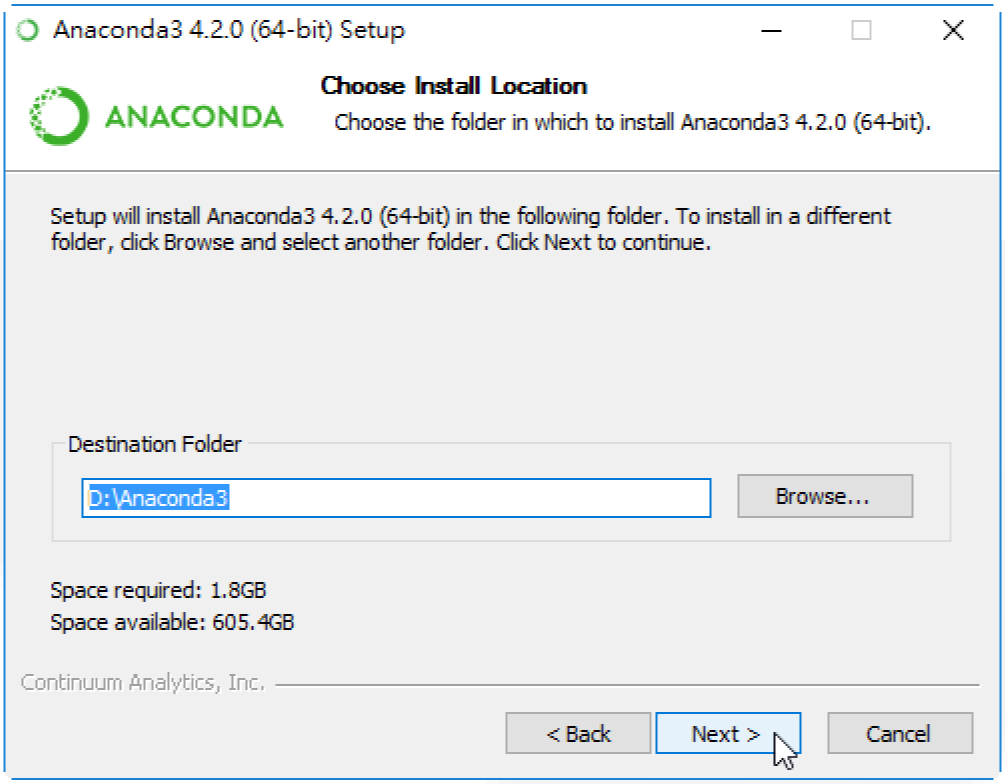
Please select the type of installation you would like to perform for Anaconda3 4.2.0 (64-bit).

Install for:

● Just Me (recommended)

○ All Users (requires admin privileges)

Continuum Analytics, Inc.



Anaconda3 4.2.0 (64-bit) Setup

Choose Install Location

Choose the folder in which to install Anaconda3 4.2.0 (64-bit).

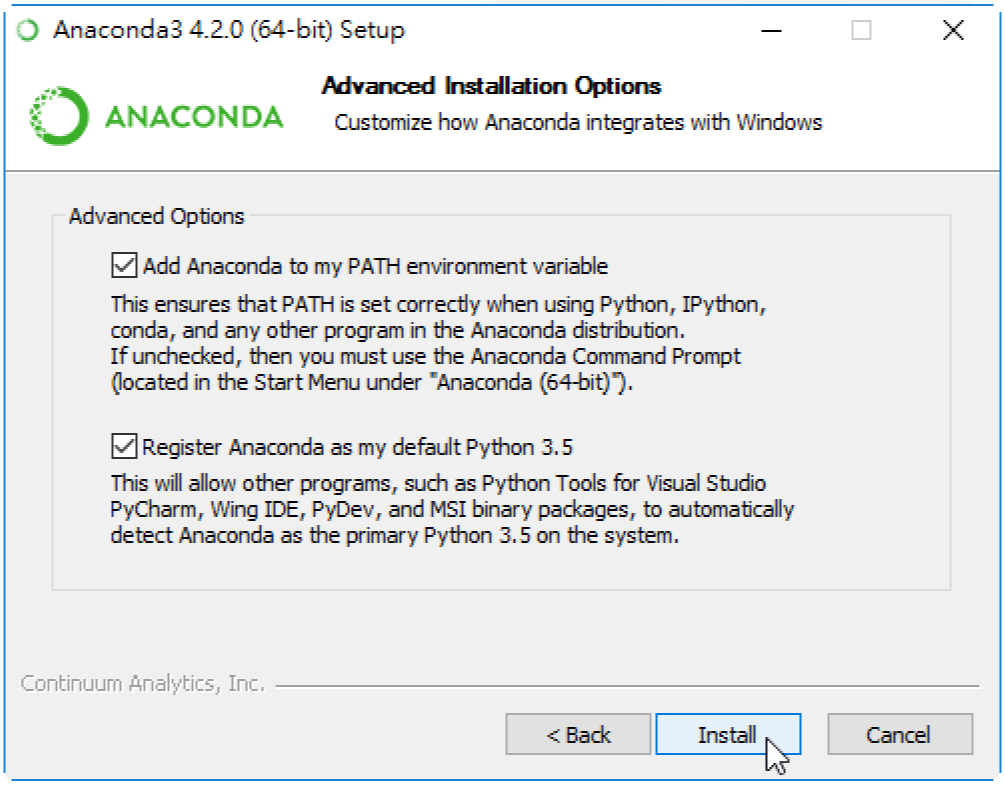
Setup will install Anaconda3 4.2.0 (64-bit) in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

Destination Folder

D:\Anaconda3

Space required: 1.8GB
Space available: 605.4GB

Continuum Analytics, Inc.



Anaconda3 4.2.0 (64-bit) Setup

Advanced Installation Options

Customize how Anaconda integrates with Windows

Advanced Options

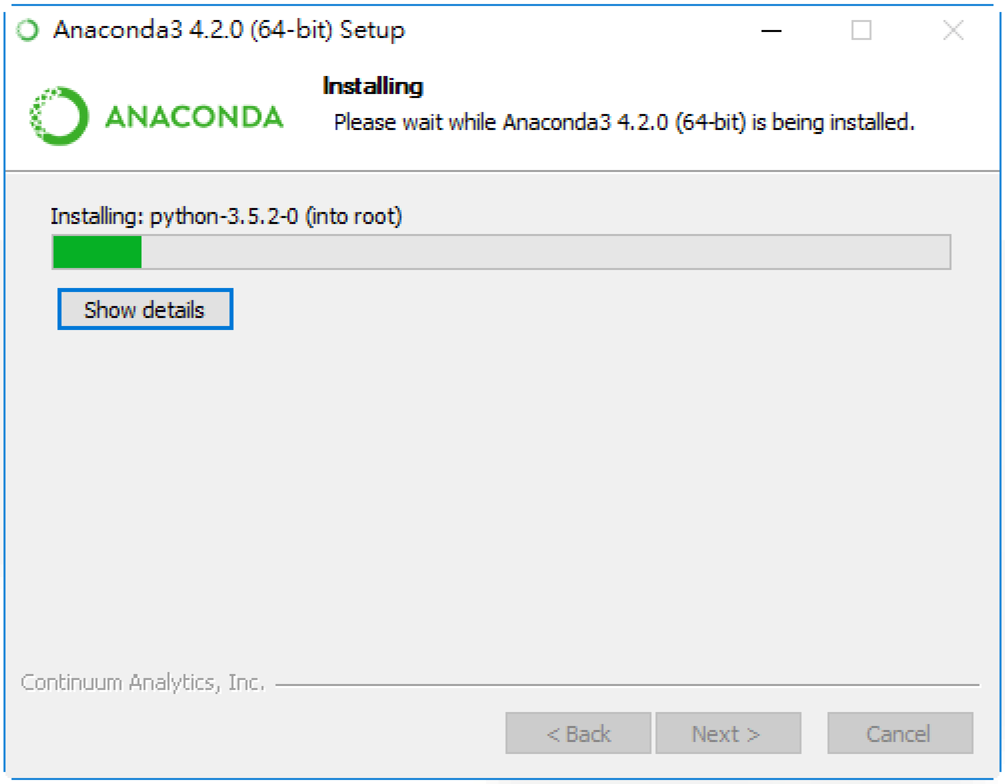
☑ Add Anaconda to my PATH environment variable

This ensures that PATH is set correctly when using Python, IPython, conda, and any other program in the Anaconda distribution.
If unchecked, then you must use the Anaconda Command Prompt (located in the Start Menu under "Anaconda (64-bit)").

☑ Register Anaconda as my default Python 3.5

This will allow other programs, such as Python Tools for Visual Studio PyCharm, Wing IDE, PyDev, and MSI binary packages, to automatically detect Anaconda as the primary Python 3.5 on the system.

Continuum Analytics, Inc.



Anaconda3 4.2.0 (64-bit) Setup

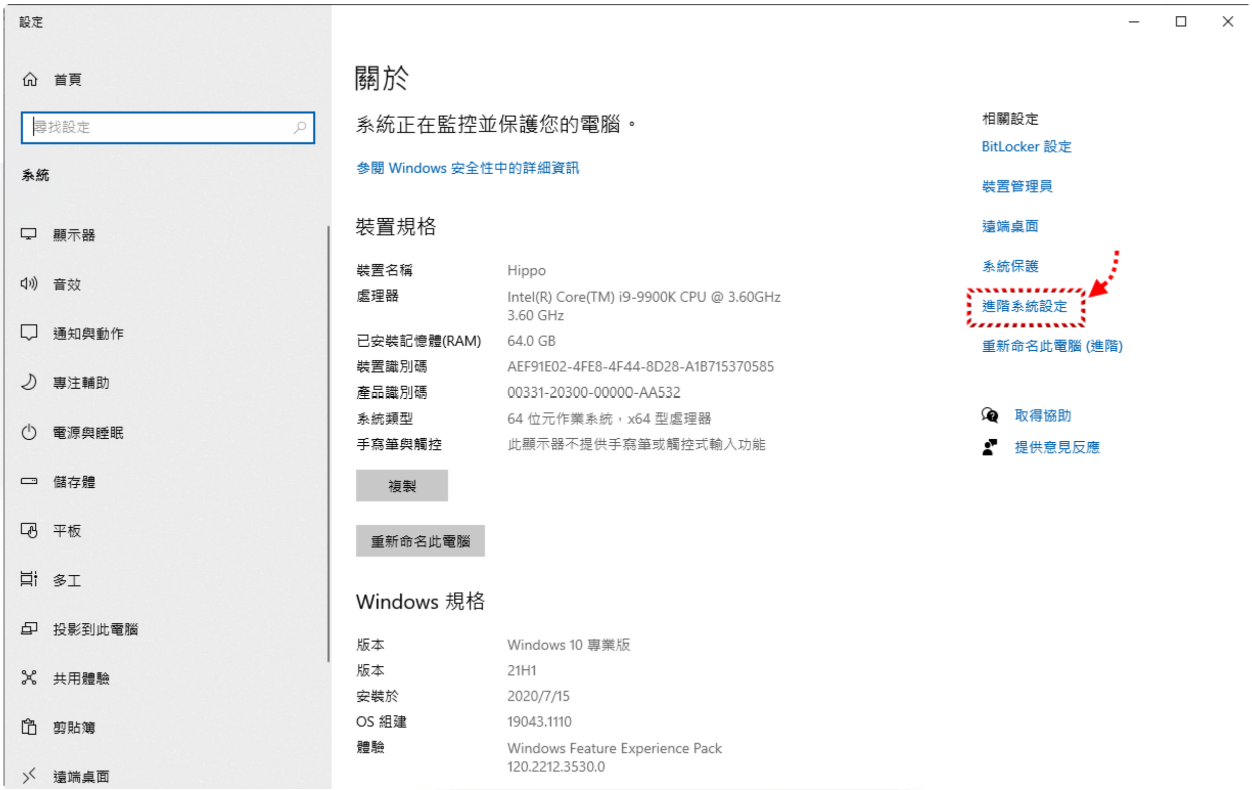
Installing

Please wait while Anaconda3 4.2.0 (64-bit) is being installed.

Installing: python-3.5.2-0 (into root)

設定環境變數

==



關於

系統正在監控並保護您的電腦。

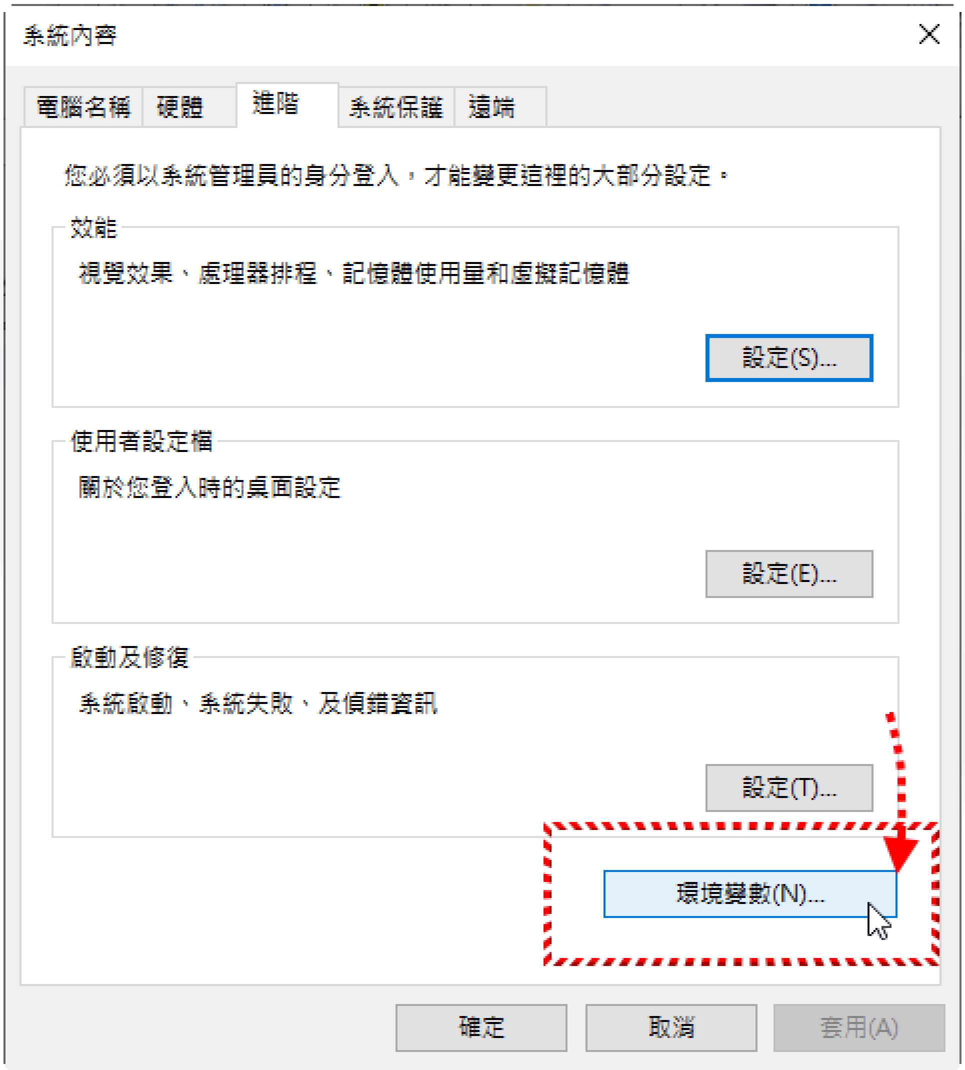
參閱 Windows 安全性中的詳細資訊

裝置規格

裝置名稱	Hippo
處理器	Intel(R) Core(TM) i9-9900K CPU @ 3.60GHz 3.60 GHz
已安裝記憶體(RAM)	64.0 GB
裝置識別碼	AEF91E02-4FE8-4F44-8D28-A1B715370585
產品識別碼	00331-20300-00000-AA532
系統類型	64 位元作業系統，x64 型處理器
手寫筆與觸控	此顯示器不提供手寫筆或觸控式輸入功能

Windows 規格

版本	Windows 10 專業版
版本	21H1
安裝於	2020/7/15
OS 組建	19043.1110
體驗	Windows Feature Experience Pack 120.2212.3530.0



系統內容

您必須以系統管理員的身分登入，才能變更這裡的大部分設定。

效能

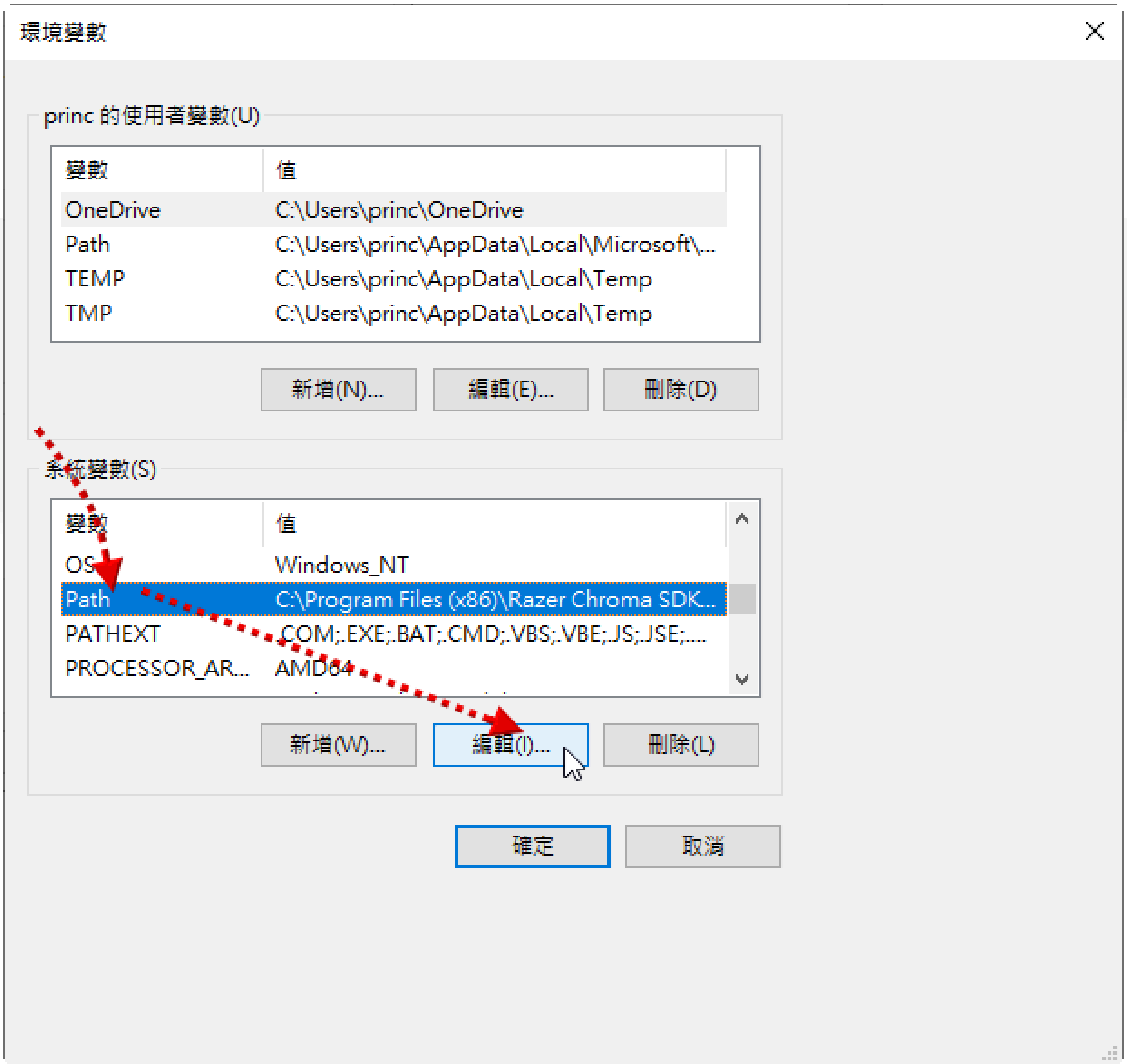
視覺效果、處理器排程、記憶體使用量和虛擬記憶體

使用者設定檔

關於您登入時的桌面設定

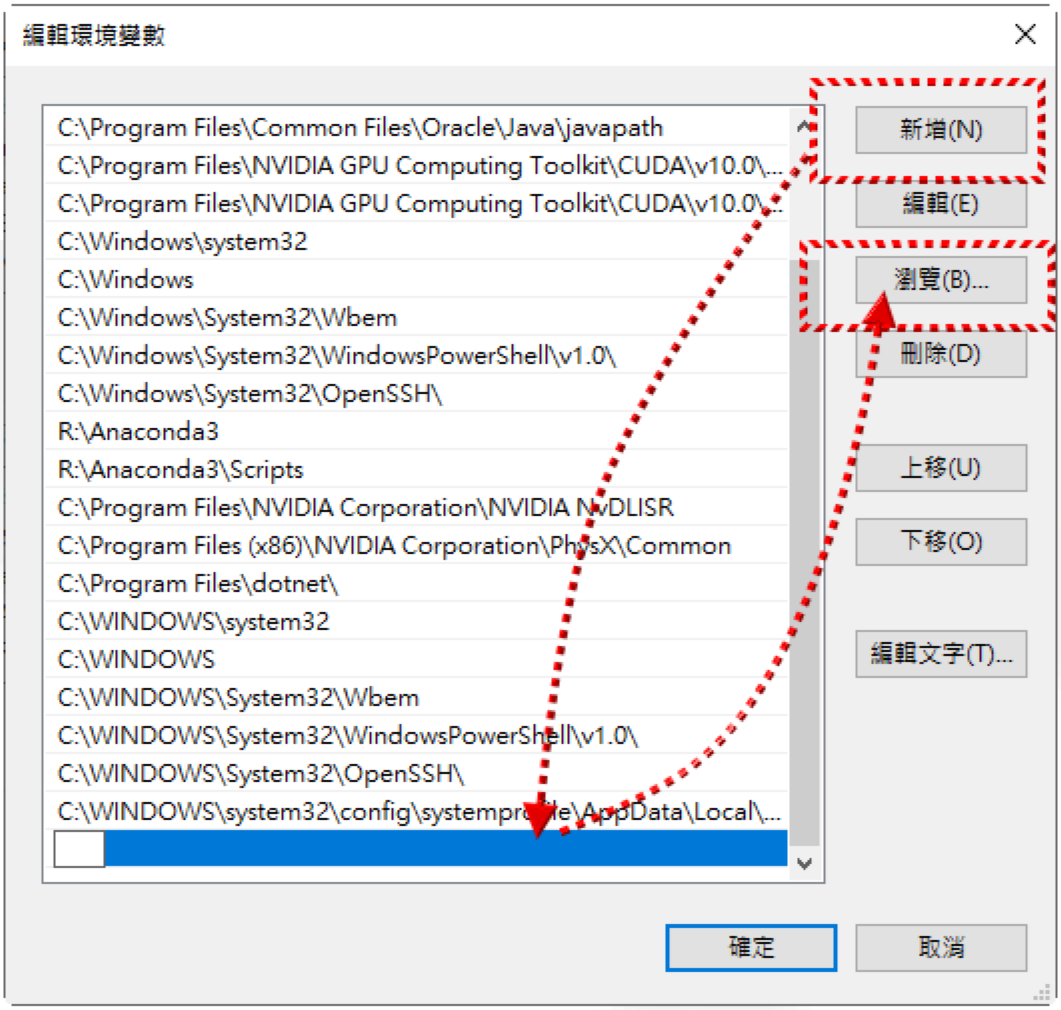
啟動及修復

系統啟動、系統失敗、及偵錯資訊



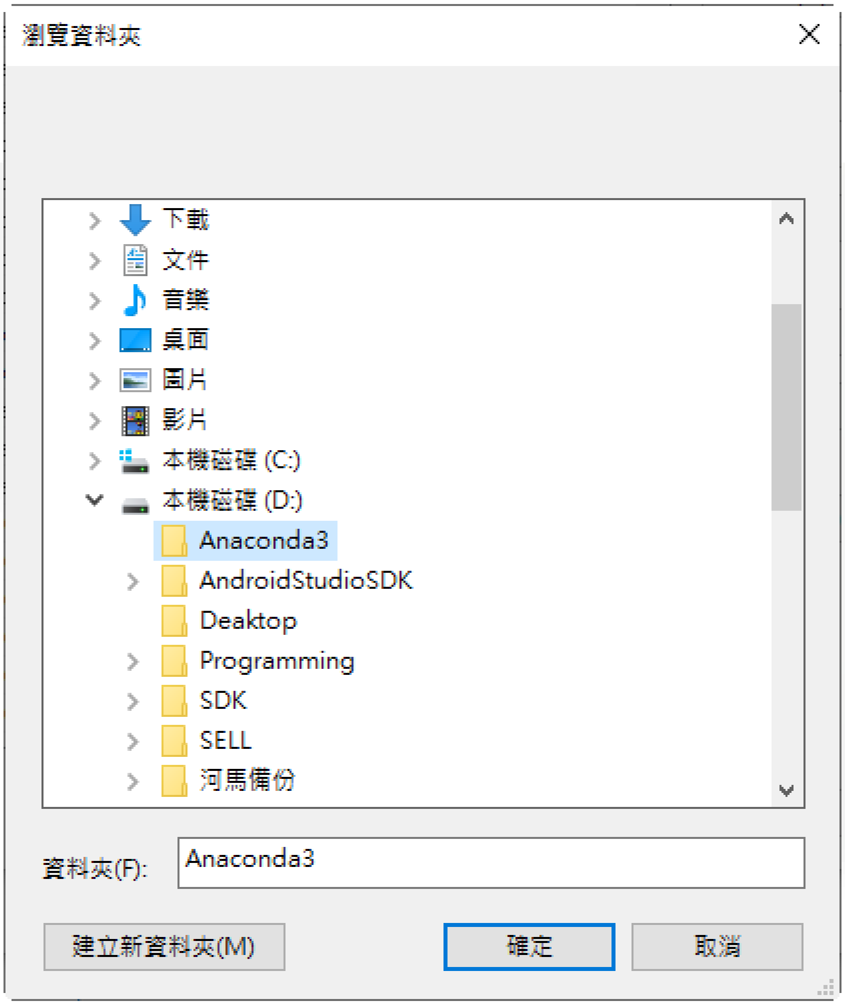
變數	值
OneDrive	C:\Users\princ\OneDrive
Path	C:\Users\princ\AppData\Local\Microsoft\...
TEMP	C:\Users\princ\AppData\Local\Temp
TMP	C:\Users\princ\AppData\Local\Temp

變數	值
OS	Windows_NT
Path	C:\Program Files (x86)\Razer Chroma SDK...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;....
PROCESSOR_AR...	AMD64



```
C:\Program Files\Common Files\Oracle\Java\javapath
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.0\...
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.0\...
C:\Windows\system32
C:\Windows
C:\Windows\System32\Wbem
C:\Windows\System32\WindowsPowerShell\v1.0\
C:\Windows\System32\OpenSSH\
R:\Anaconda3
R:\Anaconda3\Scripts
C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR
C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common
C:\Program Files\dotnet\
C:\WINDOWS\system32
C:\WINDOWS
C:\WINDOWS\System32\Wbem
C:\WINDOWS\System32\WindowsPowerShell\v1.0\
C:\WINDOWS\System32\OpenSSH\
C:\WINDOWS\system32\config\systemprofile\AppData\Local\...
```

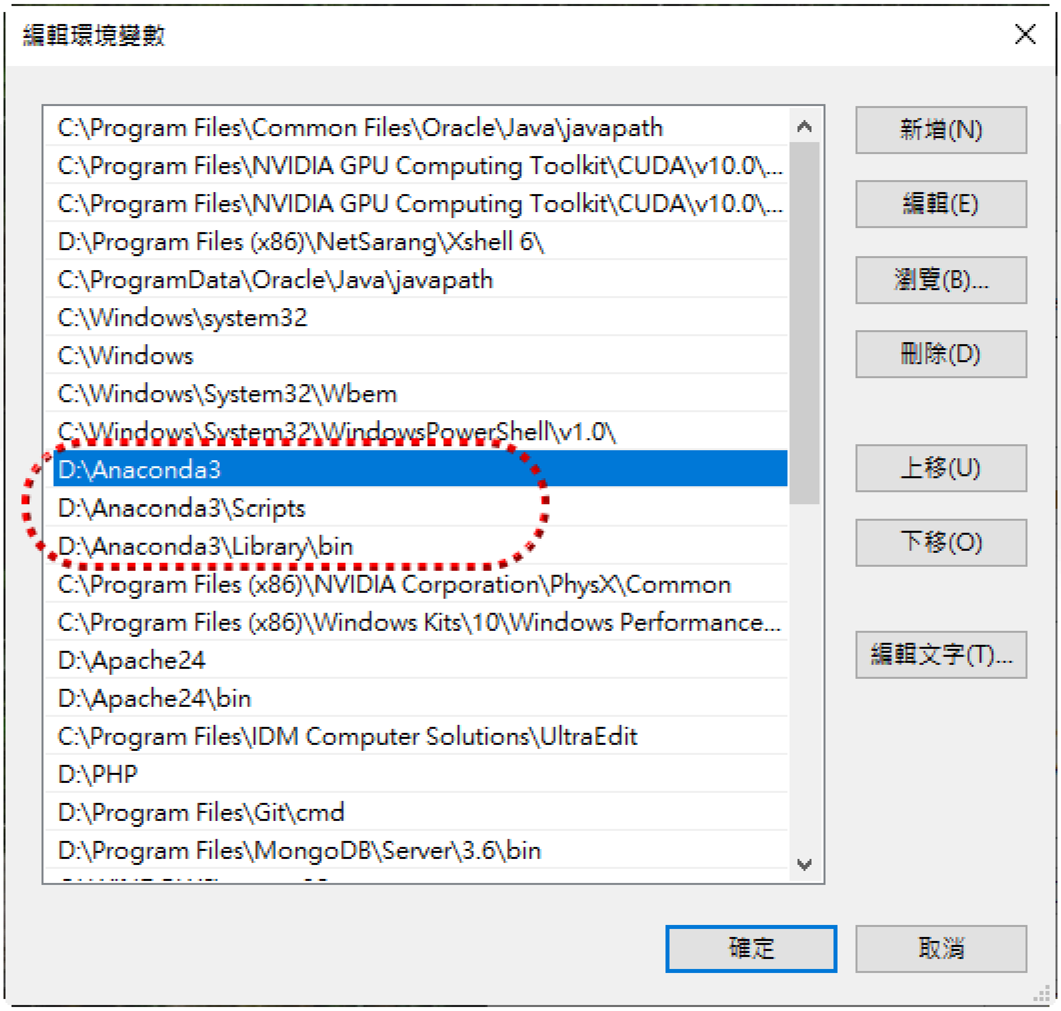
選擇剛才解壓縮的路徑 D:\Anaconda3，並按下「確定」按鈕。
===



資料夾(F): Anaconda3

除了 D:\Anaconda3，還須新增 D:\Anaconda3\Scripts 與 D:\Anaconda3\Library\bin 共三個資料夾。

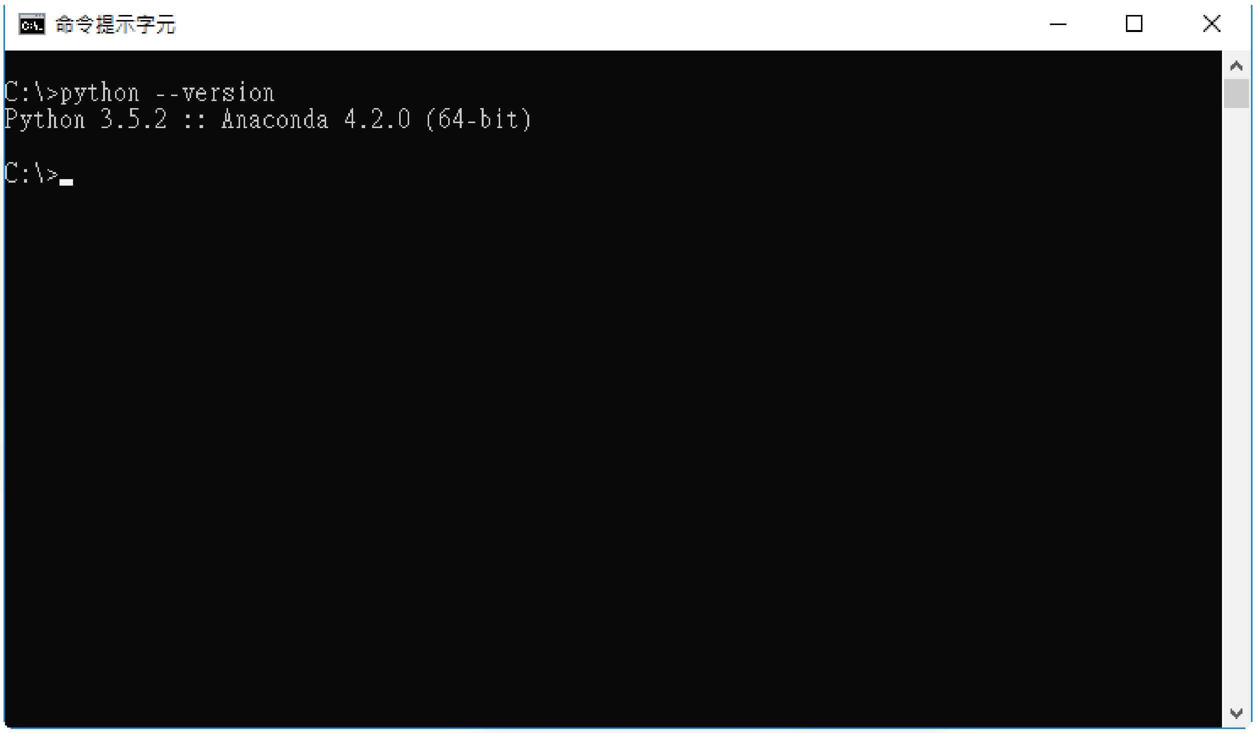
==



```
C:\Program Files\Common Files\Oracle\Java\javapath
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.0\...
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.0\...
D:\Program Files (x86)\NetSarang\Xshell 6\
C:\ProgramData\Oracle\Java\javapath
C:\Windows\system32
C:\Windows
C:\Windows\System32\Wbem
C:\Windows\System32\WindowsPowerShell\v1.0\
D:\Anaconda3
D:\Anaconda3\Scripts
D:\Anaconda3\Library\bin
C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common
C:\Program Files (x86)\Windows Kits\10\Windows Performance...
D:\Apache24
D:\Apache24\bin
C:\Program Files\IDM Computer Solutions\UltraEdit
D:\PHP
D:\Program Files\Git\cmd
D:\Program Files\MongoDB\Server\3.6\bin
```

Environment check:

1. C:\>python --version



```
C:\>python --version
Python 3.5.2 :: Anaconda 4.2.0 (64-bit)

C:\>_
```

Lesson 8 - Text pre processing, jieba, TF-IDF

Table of Contents

Source code: https://www.dropbox.com/s/8v1v9pvkni03gl8/lesson08.rar?dl=0

```
break;
```

CKIP Transformers 範例程式 (僅限支援繁體中文)
==

1. 建立一個 Notebook, 命名為 : ckip_transformer.ipynb => kernel 使用 uuunlp
2. 以下每一個 分隔符號 == 表示一個 cell

```
========

import time
from ckip_transformers.nlp import CkipWordSegmenter, CkipPosTagger, CkipNerChunker

========

start = time.time()
ws_driver = CkipWordSegmenter(device=-1) # Use CPU: -1 , Use GPU: 0
pos_driver = CkipPosTagger(device=-1)
ner_driver = CkipNerChunker(device=-1)
end = time.time()
print(end - start)

========

def extract_entities(content):
    entities = {
        "PERSON": [],
        "ORG": [],
        "GPE": [],
        "PRODUCT": [],
        "EVENT": [],
        "LAW": [],
        "LANGUAGE": [],
        "DATE": [],
        "TIME": [],
        "PERCENT": [],
        "MONEY": [],
        "QUANTITY": [],
        "ORDINAL": [],
        "CARDINAL": [],
        "NORP":[],
        "LOC":[],
        "FAC":[],
        "WORK_OF_ART":[],
    }
    ner_results = ner_driver([content])
    for result in ner_results[0]:
        text, label, idx  = result
        # print(text, label, idx)
        if text.strip() not in entities[label]:
            entities[label].append(text.strip())
    return entities

========

content = "台灣蔡阿嘎和陳大衛在台北101逛誠品，買了一個帆布包，接著去永康夜市吃牛肉麵和水餃"
entities = extract_entities(content)
print(entities)

========
def extract_ws(content):
    content = str(content)
    word_list = []
    return_list = []
    if len(content)==0:
        return ""
    ws  = ws_driver([content])
    pos = pos_driver(ws)
    ner = ner_driver([content])
    for word_ws, word_pos, word_ner in zip(ws, pos, ner):
        for y in range(len(word_ws)):
            if (len(word_ws[y])>1) and (word_ws[y] not in word_list):
                return_list.append(word_ws[y].strip())
                word_list.append(word_ws[y].strip())
    return " ".join(return_list)

==

word_list = extract_ws(content)
print(word_list)
```

jieba可接受的詞性:

—---

adj 形容詞

adv 副詞

conj 連接詞

int 感嘆詞

m 數詞 (結巴獨有)

n 名詞

o 擬聲詞

prep 介系詞,介詞

pron 代詞,代名詞

punc 標點符號

q 量詞

u 助詞,結巴獨有

unknown 未知詞

v 動詞

中研院斷詞系統：https://ckip.iis.sinica.edu.tw/service/transformers/
==
Ckip Transformers 官方網站：https://github.com/ckiplab/ckip-transformers
Ckip Tagger 官方網站：https://github.com/ckiplab/ckiptagger

CKIP Transformers 支援的詞性(POS)
===
PERSON: People, including fictional characters
ORG: Organizations, including companies, government agencies, and other groups
GPE: Geopolitical entities, including countries, cities, and regions
PRODUCT: Products, including brand names and general product categories
EVENT: Events, including natural disasters, sports events, and business events
LAW: Laws, including legal codes, regulations, and court cases
LANGUAGE: Natural languages, including English, Chinese, and other languages
DATE: Dates, including calendar dates and time periods
TIME: Times, including clock times and time periods
PERCENT: Percentage expressions, including percentages, fractions, and decimals
MONEY: Monetary expressions, including currency names, values, and financial instruments
QUANTITY: Quantities, including measurements, counts, and units
ORDINAL: Ordinal numbers, including first, second, third, etc.
CARDINAL: Cardinal numbers, including one, two, three, etc.

@ckip 斷詞時間測試
使用 100 篇中文文章
transformers:/bert-base-chinese-ws

@CPU測試
測試平台5900x
1 thread 140s 預估133.3hrs
2 thread 72s 預估 68.6 hrs
4 thread 41s 預估 39.0 hrs
8 thread 31s 預估 29.3 hrs
24 thread 27s 預估 25.7 hrs

@GPU測試
3090 跑 1000 篇花費 24s

Lesson 10 - Text Analytics & Word2Vec

Table of Contents

Source code: https://www.dropbox.com/s/78v1t8of10n3d4z/lesson10.rar?dl=0

Lesson 11 - Sentiment Analysis

Table of Contents

Source code: https://www.dropbox.com/s/6upjfar5bgs9kta/lesson11.rar?dl=0

解壓縮將:google_play_big.csv 存放在
C:\Programming\ExampleEnv\data\sentimental\googleplay\

Google Language API (付費使用 Google Language模型)
==

1) 至此下載Google Credential (你需要用有 Google Cloud Platform 帳號並啟用付款方式):
Install and Initial GCP SDK: https://cloud.google.com/sdk/docs/install

2) 建立虛擬環境

```
conda create -n uuugoogle python=3.7
conda activate uuugoogle
pip install ipykernel
python -m ipykernel install --user --name uuugoogle --display-name uuugoogle
pip install ipywidgets widgetsnbextension pandas-profiling
pip install --upgrade google-cloud-language
```

ref: https://cloud.google.com/python/docs/reference/language/latest

3) 進入虛擬環境

```
conda activate uuugoogle
(uuugoogle) C:\Programming\ExampleEnv>jupyter notebook
```

LAB: 大衛的中文情感模型 (請按 follow 追隨)
https://huggingface.co/DavidLanz

=================================

建立一個 chinese_sentiment.ipynb 檔案，依序貼上cell

```
import torch
from transformers import BertForSequenceClassification
from transformers import BertTokenizer
```

==

```
# 下載大衛的model from HuggingFace
tokenizer =BertTokenizer.from_pretrained('DavidLanz/fine_tune_chinese_sentiment')
model =
BertForSequenceClassification.from_pretrained('DavidLanz/fine_tune_chinese_sentiment')
```

==

```
text='阿不就好棒棒'
output = model(torch.tensor([tokenizer.encode(text)]))
print(torch.nn.functional.softmax(output.logits,dim=-1))
```

==

```
class_label = {
    0:'Semi-negation',
    1:'Negation',
    2:'Neutral',
    3:'Semi-positive',
    4:'Positive',
}

def argsort(seq):
    return sorted(range(len(seq)), key=seq.__getitem__)

def predict_sentiment(model, tokenizer, sentence):
    input_ids = torch.tensor([tokenizer.encode(sentence)])
    pred_list = []
    ps = []
    return_dict = {}
    with torch.no_grad():
        out = model(input_ids)
        pred_list = out.logits.softmax(dim=-1).tolist()
    top5 = argsort(pred_list[0])[-5:][::-1]
    for i in top5:
        ps.append({class_label[i]:pred_list[0][i]})
    return class_label[top5[0]], ps, sentence

==
text='酸民的話語太狠了'
text='阿不就好棒棒'

predict, ps, sentence = predict_sentiment(model, tokenizer, text)
print(predict, ps, sentence)

==

sorted_data = sorted(ps, key=lambda x: list(x.values())[0], reverse=True)

highest_key = list(sorted_data[0].keys())[0]
highest_value = list(sorted_data[0].values())[0]
print(sorted_data[0])
```

```
break;
```

Lesson 21 - Google Play BERT Sentiment Classifier

Table of Contents

Source code: https://www.dropbox.com/s/a1l1mjizb2m93re/lesson21.rar?dl=0

Source code:
https://colab.research.google.com/drive/1OTcTZ6Yctj73oSZUf9Wq0q_RiJTGQHWJ?usp=sharing

Lesson 22 - Fine-tuning RNN

Table of Contents

Source code: https://www.dropbox.com/s/dcol2m3bxaqdoea/lesson22.rar?dl=0

Google Colab:
https://colab.research.google.com/drive/1xTWLk_M56fCvR71D8QxDCUFRIyU7k8eH?usp=sharing

Multi-Class, Multi-Label 多類別文章分類與標籤文章分類

==

多類別文章分類和多標籤文章分類的的區別有二，分別為Multi-Class與Multi-Label。

Multi-Class：多類別/多元分類（二分類、三分類、多分類等）

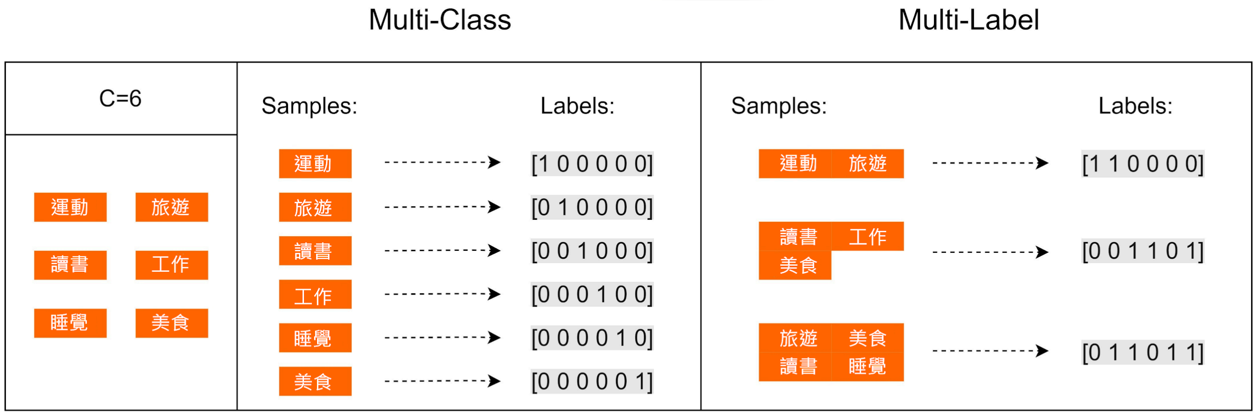
二分類：判斷郵件屬於哪個類別，垃圾或者非垃圾
二分類：判斷新聞屬於哪個類別，機器寫的或者人寫的
三分類：判斷文本情感屬於{正面，中立，負面}中的哪一類
多分類：判斷新聞屬於哪個類別，如財經、體育、娛樂等

Multi-Label：多標籤分類

文本可能同時涉及任何宗教，政治，金融或教育，也可能不屬於任何一種。
電影可以根據其摘要內容分為動作，喜劇和浪漫類型。有可能電影屬於 romcoms (浪漫與喜劇)等多種類型。
二者區別

多分類任務中一條資料只有一個標籤，但這個標籤可能有多種類別。比如判定某個人的性別，只能歸類為"男性"、"女性"其中一個。再比如判斷一個文本的情感只能歸類為"正面"、"中面"或者"負面"其中一個。
多標籤分類任務中一條資料可能有多個標籤，每個標籤可能有兩個或者多個類別（一般兩個）。例如，一篇新聞可能同時歸類為"娛樂"和"運動"，也可能只屬於"娛樂"或者其它類別。
舉例：

假設個人愛好的集合一共有6個元素：運動、旅遊、讀書、工作、睡覺、美食



Lesson 17 - Keyword Extraction by TextRank using spacy

Table of Contents

```
C:\Programming\ExampleEnv>conda activate uuunlp
(uuunlp) C:\Programming\ExampleEnv>pip install spacy==3.4.1
(uuunlp) C:\Programming\ExampleEnv>python -m spacy download en_core_web_sm
(uuunlp) C:\Programming\ExampleEnv>python -m spacy download zh_core_web_sm
(uuunlp) C:\Programming\ExampleEnv>pip install plac
```

https://colab.research.google.com/drive/19xvrZKfCygIHayQTVjIcKTtdG32tRj6q?usp=sharing

```
break;
```

Lesson 12 - Gender Prediction Based on Name

Table of Contents

Source code: https://www.dropbox.com/s/v3fueks6oga1pe0/lesson12.rar?dl=0

Lesson 15 - Update Jieba or Ckip Chinese dictionary

Table of Contents

Source code: https://www.dropbox.com/s/wo28atnwldbz3yd/lesson15.rar?dl=0

n=1 : Unigram Model
C是指文字 i 出現的次數，M是指文集中所有字數。

$$P(w_i) = \frac{C(w_i)}{M}$$

n=2 : Bigram model

$$P(w_i | w_{i-1}) = \frac{C(w_{i-1} w_i)}{C(w_{i-1})}$$

n=3 : Trigram model

$$P(w_i | w_{i-n+1} \ldots w_{i-1}) = \frac{C(w_{i-n+1} \ldots w_i)}{C(w_{i-n+1} \ldots w_{i-1})}$$

下載：Google Gemma (Gemini gemma-3-4b-it-GGUF)
https://huggingface.co/lmstudio-community/gemma-3-4b-it-GGUF/resolve/main/gemma-3-4b-it-Q4_K_M.gguf?download=true

下載：Google Gemma (Gemini gemma-2-2b-it-GGUF)
https://huggingface.co/lmstudio-community/gemma-2-2b-it-GGUF/resolve/main/gemma-2-2b-it-Q4_K_M.gguf?download=true

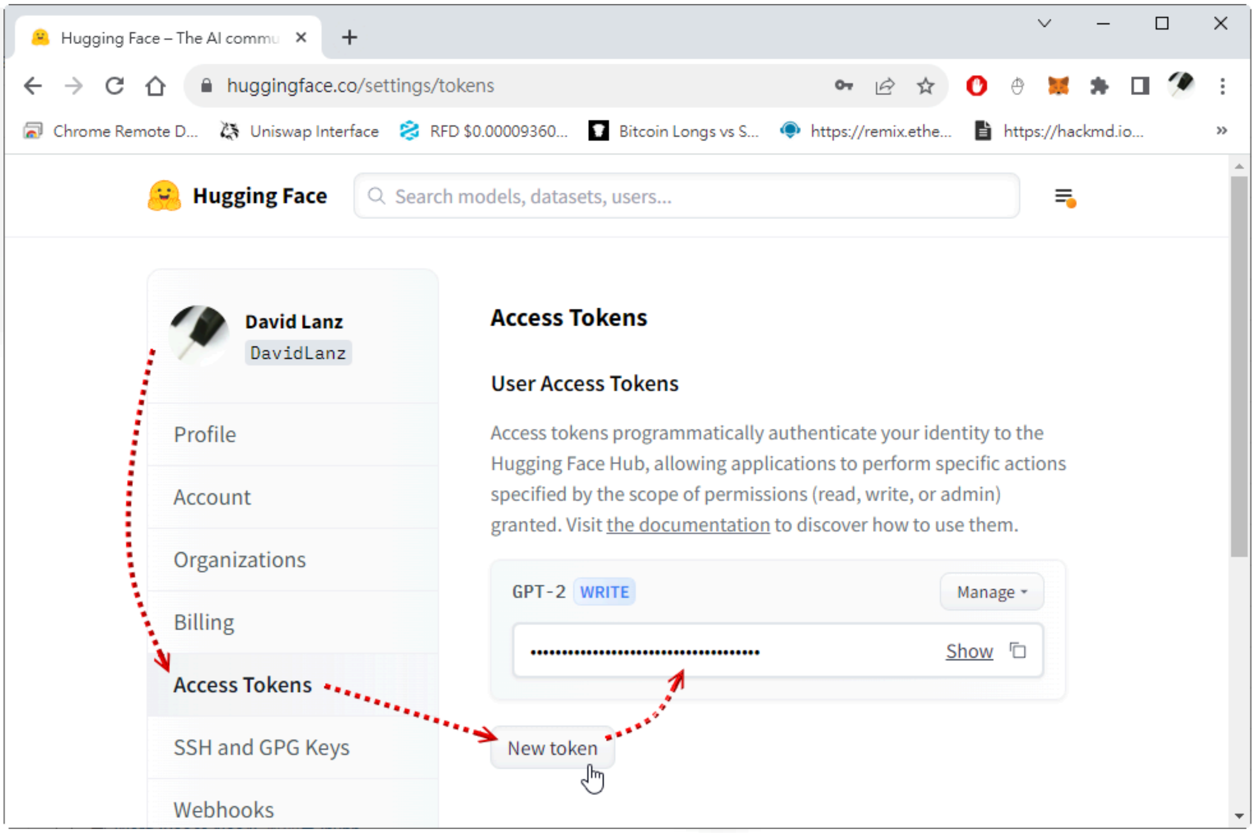
下載聯發科模型 (Breeze-7B-Instruct-v0.1-Q4_K_M.gguf)：
https://huggingface.co/audreyt/Breeze-7B-Instruct-v0.1-GGUF/resolve/main/Breeze-7B-Instruct-v0.1-Q4_K_M.gguf?download=true

下載 中文 Embedding 模型 (bge-large-zh-v1.5-q4_k_m.gguf)：
https://huggingface.co/CompendiumLabs/bge-large-zh-v1.5-gguf/resolve/main/bge-large-zh-v1.5-q4_k_m.gguf?download=true

HuggingFace

===

申請與建立 HuggingFace 帳號：https://huggingface.co/
點選自己的大頭照，選擇 Access Token，按下：New token
複製 HfApi Access Token

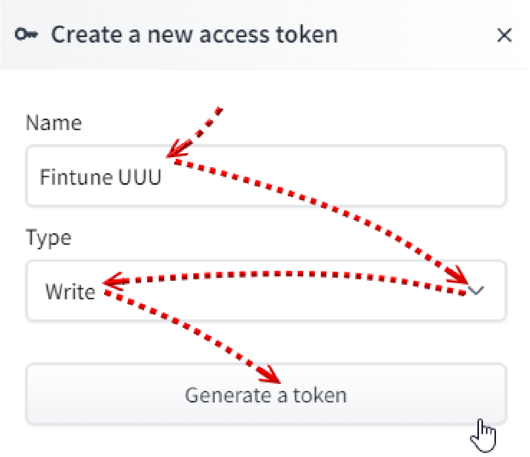


Access Tokens

User Access Tokens

Access tokens programmatically authenticate your identity to the Hugging Face Hub, allowing applications to perform specific actions specified by the scope of permissions (read, write, or admin) granted. Visit the documentation to discover how to use them.

下拉 Type 選單，權限為 Write 可寫入，課程稍後會使用此 Token 來上傳模型。

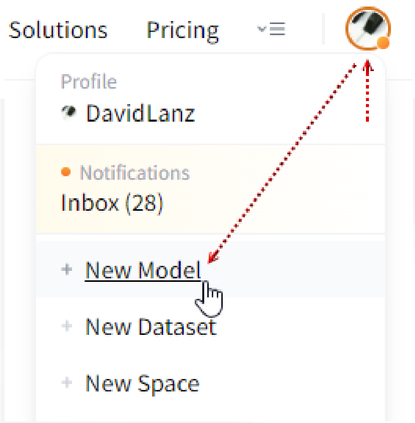


Create a new access token

Name: Fintune UUU

Type: Write

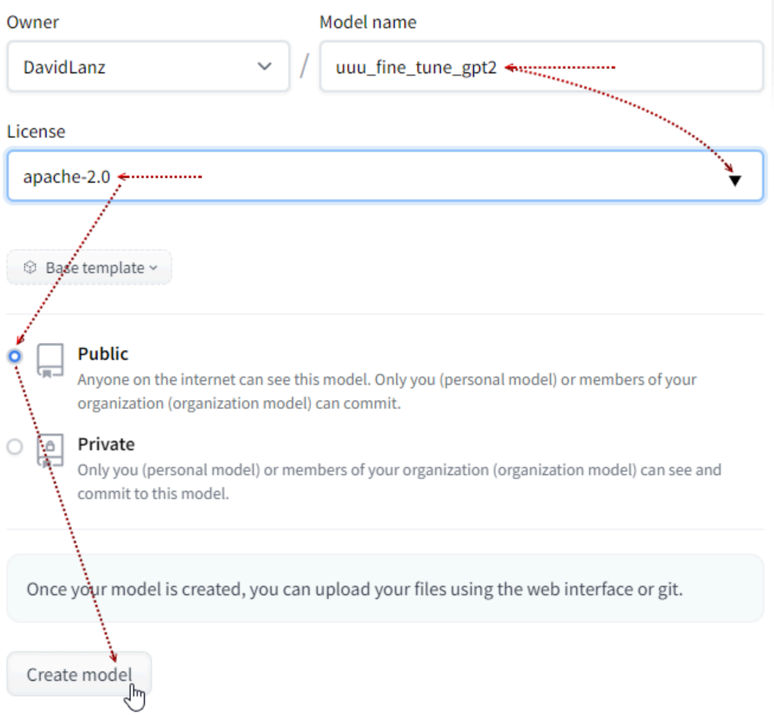
至 Hugging Face 上登入帳號後，建立以下四個名稱的模型
==



名稱：

- tcp2023
- uuu_fine_tune_taipower
- uuu_fine_tune_gpt2
- llama2_uuu_news_qlora

分別建立四個模型，模型授權參考，建議都選擇 Apache 2.0，如下圖所示：



Owner: DavidLanz / Model name: uuu_fine_tune_gpt2

License: apache-2.0

● **Public**
Anyone on the internet can see this model. Only you (personal model) or members of your organization (organization model) can commit.

○ **Private**
Only you (personal model) or members of your organization (organization model) can see and commit to this model.

Once your model is created, you can upload your files using the web interface or git.

Apache 2.0：最常用的授權，允許自由使用、修改和分發模型，但您必須保留原始作者的版權聲明。

MIT：這是另一個常用的授權，允許自由使用、修改和分發模型，但您不需要保留原始作者的版權聲明。

GPL：這是一個自由軟體授權，允許自由使用、修改和分發模型，但您必須將任何修改的版本以 GPL 授權重新發佈。

Day break;

Morning!

Review
==

1. What is the difference between supervised and unsupervised learning?
2. Can you create a pair plot for each feature in a given training dataset?
3. There are several methods for converting words into vectors: WordNet, one-hot encoding, and Word2Vec. Could you explain when you would choose each method in different scenarios?
4. How does TF-IDF work, particularly in the context of transforming documents into vectors?
5. Do you have a Hugging Face account?
6. How do you prepare a dataset for a multi-label model? Is it similar to preparing a sentiment dataset?
7. How can I extract keywords from an HTML web page, and can these keywords be used as a dictionary in Jieba?
8. What are the differences between Jieba and CKIP Transformers for Chinese word segmentation?

Note:
Before starting today's session, please copy these models from David's USB drive. (Repeat: DO NOT download them directly.)

下載：Google Gemma (Gemini gemma-3-4b-it-GGUF)
https://huggingface.co/lmstudio-community/gemma-3-4b-it-GGUF/resolve/main/gemma-3-4b-it-Q4_K_M.gguf?download=true

下載： Google Gemma (Gemini gemma-2-2b-it-GGUF)
https://huggingface.co/lmstudio-community/gemma-2-2b-it-GGUF/resolve/main/gemma-2-2b-it-Q4_K_M.gguf?download=true

下載聯發科模型 (Breeze-7B-Instruct-v0.1-Q4_K_M.gguf)：
https://huggingface.co/audreyt/Breeze-7B-Instruct-v0.1-GGUF/resolve/main/Breeze-7B-Instruct-v0.1-Q4_K_M.gguf?download=true

下載 中文 Embedding 模型 (bge-large-zh-v1.5-q4_k_m.gguf)：
https://huggingface.co/CompendiumLabs/bge-large-zh-v1.5-gguf/resolve/main/bge-large-zh-v1.5-q4_k_m.gguf?download=true

大型語言模型 Large Language Model

1. 註冊 ChatGPT 帳號：https://chatgpt.com/

2. 註冊 Google Gemini 帳號：https://gemini.google.com/

3. 註冊 Claude 帳號：https://claude.ai/

4. 註冊 Grog 帳號: https://groq.com/

```
break;
```

Lesson 20 - Fine-Tune GPT-2 for Text Generation

Table of Contents

Source Code:
https://colab.research.google.com/drive/1USNGRzSe21fQAVzSMtNV1IvV05pY2Lq-?usp=sharing

注意：此Notebook需要使用Colab的GPU，如果訓練完成，請在右上方instance的下拉選單，選擇：「中斷連線並刪除執行階段」，將GPU歸還給Google。

Google Colab T4 的使用限制
===

Google Colab 對免費版的 GPU 使用有嚴格的限制。如果您定期使用免費版的 GPU，您的可執行時間會逐漸縮短，斷線的頻率會增加。您還可能需要等待更長的時間才能重新連接到 GPU。

Google Colab 會監控您的使用情況，並可能因過度使用而對您的帳戶實施限制。他們不會明確告訴您您的帳戶被限制的原因，也不會為您提供追蹤您的使用情況的方法。

對於 Colab Pro 用戶，Google Colab 的限制較少，但仍可能會因過度使用而對您的帳戶實施限制。

總而言之，Google Colab 是一個免費的服務，但並不適合長期的執行。如果您需要使用 GPU 進行大量的計算，您應該考慮付費的 Colab Pro 或其他雲端服務。

===

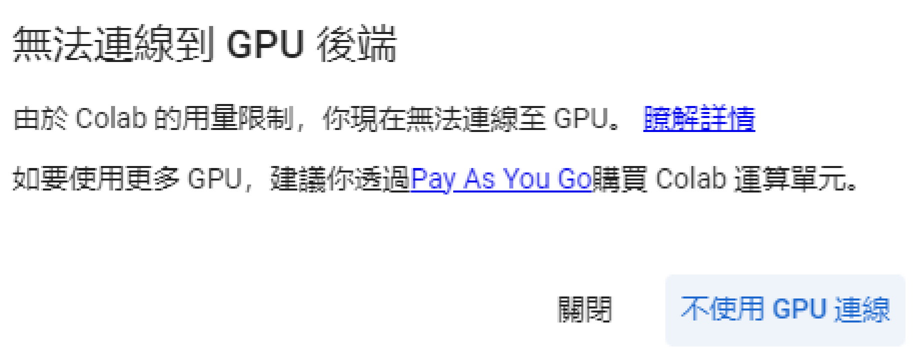
如果有一天，你看見了這些文字...

===

"由於 Colab 的用量限制，你現在無法連線至 GPU。瞭解詳情
如要使用更多 GPU，建議你透過Pay As You Go購買 Colab 運算單元。"

"Unable to connect to GPU backend
You cannot currently connect to a GPU due to usage limits in Colab. More information
If you want more access to GPUs, you can buy Colab compute units with Pay As You Go."

https://research.google.com/colaboratory/faq.html

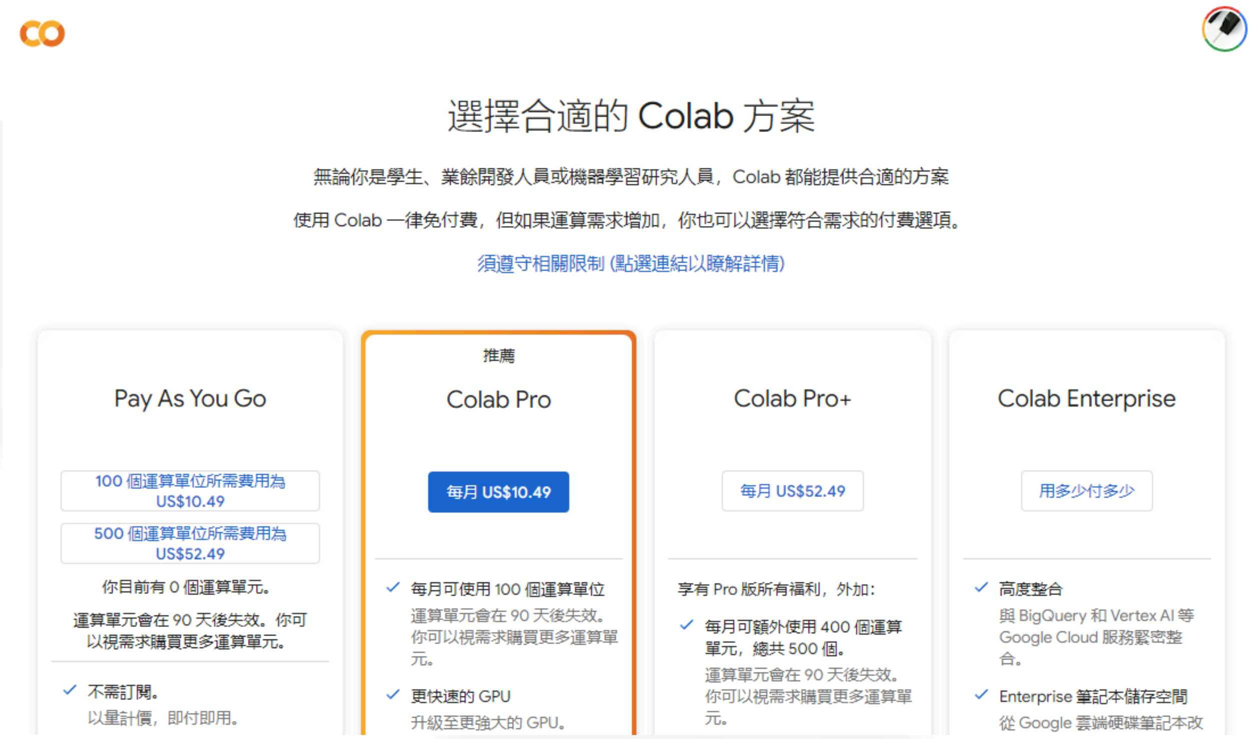


無法連線到 GPU 後端

由於 Colab 的用量限制，你現在無法連線至 GPU。 瞭解詳情

如要使用更多 GPU，建議你透過Pay As You Go購買 Colab 運算單元。

如果你一直無法使用免費的 T4，可選購採用Pay As You Go 使用多少付多少來做練習。或者建議訂閱每個月10.49 (台幣342元/月)，可以使用到 A100 的大型語言模型訓練。



選擇合適的 Colab 方案

無論你是學生、業餘開發人員或機器學習研究人員，Colab 都能提供合適的方案

使用 Colab 一律免付費，但如果運算需求增加，你也可以選擇符合需求的付費選項。

須遵守相關限制 (點選連結以瞭解詳情)

Pay As You Go

100 個運算單位所需費用為 US$10.49

500 個運算單位所需費用為 US$52.49

你目前有 0 個運算單元。

運算單元會在 90 天後失效。你可以視需求購買更多運算單元。

✓ 不需訂閱。
以量計價，即付即用。

推薦

Colab Pro

每月 US$10.49

✓ 每月可使用 100 個運算單位
運算單元會在 90 天後失效。你可以視需求購買更多運算單元。

✓ 更快速的 GPU
升級至更強大的 GPU。

Colab Pro+

每月 US$52.49

享有 Pro 版所有福利，外加：

✓ 每月可額外使用 400 個運算單元，總共 500 個。
運算單元會在 90 天後失效。你可以視需求購買更多運算單元。

Colab Enterprise

用多少付多少

✓ 高度整合
與 BigQuery 和 Vertex AI 等 Google Cloud 服務緊密整合。

✓ Enterprise 筆記本儲存空間
從 Google 雲端硬碟筆記本改

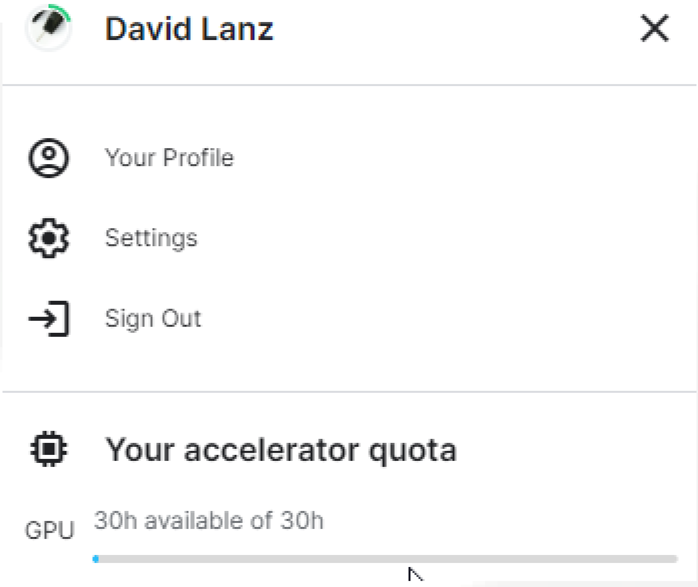
Google Colab 免費或付費版本的每個工作階段，都有最長 24 小時的 GPU 執行時間限制。此限制適用於所有 GPU 類型，包括 NVIDIA P100 和 T4 GPU。

如果超出了 GPU 執行時間限制，Colab Notebook 將與 GPU 一起重新啟動工作階段。這意味著您需要重新啟動 Notebook 才能重新連接 GPU 的存取權限。

以下是一些避免超出 GPU 執行時間限制的方法：

<> 只在需要時使用 GPU。如果執行的Notebook不需要 GPU 密集訓練，請中斷 GPU 連結。
<> 訓練模型時使用較小的批次大小和較少的訓練週期。這可以減少總體訓練時間和 GPU 使用。
<> 訓練大型語言模型，請使用 Cloud TPU 而非 GPU，但成本高。
<> 若需要執行 Colab 超過 24 小時，可以考慮在 GCP 市場購買VM，具靈活性，但也比 Colab Notebook 更昂貴。

免費的 Kaggle全新的29GB RAM，16GB顯示卡
==
Kaggle 升級了他們的免費套餐到 P100，提供了 29GB 記憶體和 4 個 CPU 核心，1 GPU 15.9GB VRAM。訓練限制於每週 30 小時，對於 GPU 資源有限的人來說仍然很有吸引力。



官方網站：https://www.kaggle.com/

新聞出處：https://www.kaggle.com/discussions/product-feedback/448251

實體的獨立顯示卡 Anaconda 環境建置 PyTorch GPU 本地下載，以 RTX 40 系列為例
==
CUDA：12.4
O.S.：Windows 11

1. 下載 CUDA SDK：
https://developer.nvidia.com/cuda-12-4-0-download-archive

2. 下載 CUDNN
cudnn-windows-x86_64-8.9.7.29_cuda12-archive.zip
https://developer.nvidia.com/rdp/cudnn-archive

3. 解壓縮至資料夾：
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v12.4

4. 於虛擬環境中，安裝 PyTorch 2.6.0 GPU 版本

```
pip install torch==2.6.0 torchvision torchaudio --index-url https://download.pytorch.org/whl/cu124
```

測試：

```
import torch
torch.cuda.is_available()
```

Ubuntu CUDA 環境安裝（CUDA 12.4）
===

<> 安裝 NVIDIA 顯示卡驅動程式（建議使用官方 PPA）

```
sudo add-apt-repository ppa:graphics-drivers/ppa
sudo apt update
sudo ubuntu-drivers autoinstall
sudo reboot
```

可使用 nvidia-smi 檢查驅動是否安裝成功。

<> 安裝 CUDA 12.4（不建議使用 .run，改用 .deb）

```
# 下載並安裝 CUDA 12.4（以 Ubuntu 22.04 為例）
wget https://developer.download.nvidia.com/compute/cuda/12.4.1/local_installers/cuda-repo-ubuntu2204-12-4-local_12.4.1-1_amd64.deb
sudo dpkg -i cuda-repo-ubuntu2204-12-4-local_12.4.1-1_amd64.deb
sudo cp /var/cuda-repo-ubuntu*/cuda-*-keyring.gpg /usr/share/keyrings/
sudo apt-get update
sudo apt-get -y install cuda
```

<> 安裝 cuDNN（需登入 NVIDIA 官方帳號下載）

假設你下載的是 cuDNN 8.9.x for CUDA 12.4 的 .tar.xz 檔案：

```
tar -xvf cudnn-linux-x86_64-8.9.*_cuda12-archive.tar.xz
cd cudnn-linux-x86_64-8.9.*_cuda12-archive

# 複製 cuDNN 至 CUDA 路徑（需使用 root 權限）
sudo cp include/cudnn*.h /usr/local/cuda/include
sudo cp lib/libcudnn* /usr/local/cuda/lib64
sudo chmod a+r /usr/local/cuda/include/cudnn*.h /usr/local/cuda/lib64/libcudnn*
```

<> 設定環境變數（建議加到 ~/.bashrc）

```
echo 'export PATH=/usr/local/cuda/bin:$PATH' >> ~/.bashrc
echo 'export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH' >> ~/.bashrc
source ~/.bashrc
```

<> 建立 Conda 環境並安裝 PyTorch（GPU 支援 CUDA 12.4）

```
# 建立並進入新的 conda 環境
conda create -n torch_gpu python=3.10 -y
conda activate torch_gpu

# 安裝 PyTorch（支援 CUDA 12.4）
pip install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu124
```

```
# 測試 GPU 是否可用
import torch
print(torch.cuda.is_available())  # 應為 True
print(torch.cuda.get_device_name(0))  # 應顯示 RTX 顯示卡型號
```

```
break;
```

Lesson 29 - Retrieval-Augmented Generation with OpenAI

Table of Contents

Source code:
https://colab.research.google.com/drive/1R9rHCCNFl6SulRJTVUBiuzOzZLh5tzHE?usp=sharing

Lesson 33 - Google Gemini-Pro + Langchain + RAG

Table of Contents

Source code:
https://colab.research.google.com/drive/1tSIjzADbHkvgCKXNeF_UhAI1L1b8w4AV?usp=sharing

Lesson 40 - LlamaIndex Splitter 分片資料的比較

Table of Contents

Source code:
https://colab.research.google.com/drive/1LbH_Lf-LseuNWdFZTyFN6SfDjgLVgYm0?usp=sharing

Bonus round : Create GPTs on OpenAI
==

LnData 美妝產業知識家：
https://chatgpt.com/g/g-R7hij6YUd-lndata-mei-zhuang-chan-ye-zhi-shi-jia

練習：2023小資女孩最喜歡的開架式化妝品有哪些?

1. New GPTs

2, Create
說明：
–
建立一個能閱讀網址內容的GPT，並且根據新聞內容總結資料，分析其觀點。
(Reads and summarizes web content, analyzing viewpoints.)

指令：
–
Create an action where users can request information about a specific URL. When a user inputs the url, your action should utilize the designated API to retrieve details about that content. Currently, the API response includes only the content of the news. Ensure that your action captures this information accurately and presents it back to the user. Keep in mind that the information provided to users is limited to what is available from the API's response. There is a reference link in the content, please do append in the response body, and translate the response to Traditional Chinese. Then send a summary to call send_line API.

取消：網頁瀏覽 checkbox

3. https://www.webpilot.ai/post-gpts/

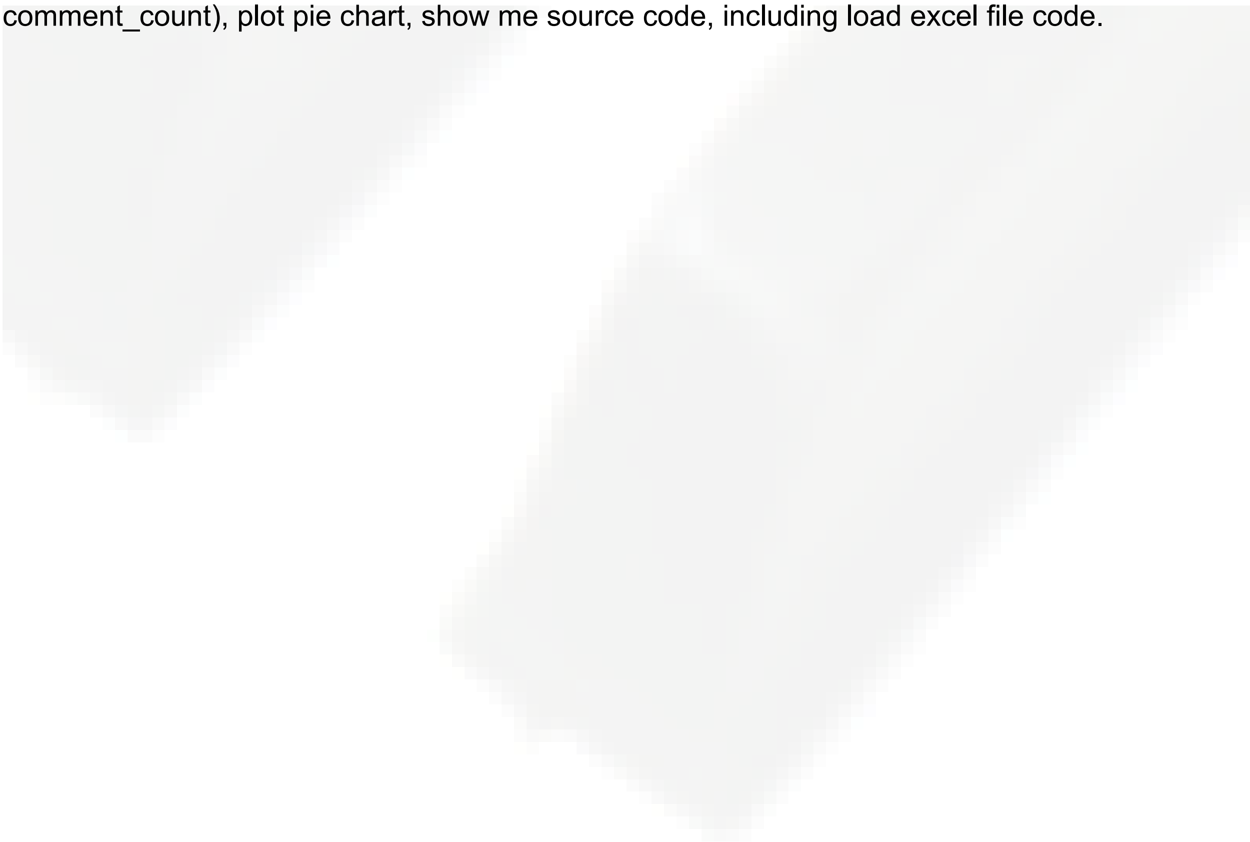
4. AWS Lambda Function

```
openapi: 3.1.0
info:
  title: ETToday API
  description: Return the details of News content
  version: 1.0.0
servers:
  - url: https://somewhere/endpoint
paths:
  /default/site_crawler_action:
    get:
      description: Return the details of a ETToday specified by the content
      operationId: get_ettoday
```

```
parameters:
  - name: url
    schema:
      type: string
    in: query
    required: true
    description: URL of a news article
```

https://colab.research.google.com/drive/1YwNvcWMM03ciZS6yjW2MzaY8far5wfJU?usp=sharing

file name: results_20240827_170853.xlsx with sheet name: FB, IG, PTT, Forum, Dcard, YouTube, News, extract posts count for each platform, and interactions (column name: comment_count), plot pie chart, show me source code, including load excel file code.



```
break;
```