



# 40 分鐘簡單聊聊 NLP

講者：Andy Chiang



# 自我介紹

- **Andy Chiang (江尚軒)**
- 中興大學資工系 大三升大四
- 中興大學 NLP 實驗室 研究助理
- 工研院 資料服務與智慧決策部 實習生
- 主要研究領域有：網頁前後端、機器學習和自然語言處理



# 開始之前...

今天的投影片有公開，大家可以掃描 QR code，待會聽演講時可以參考。

# 目錄

1

What? NLP 簡介

2

Why? NLP 實際應用

3

When? NLP 發展史

4

How?  
Google Colab +  
Hugging Face 實作

5

Reference

6

Summary

1

# What? NLP 簡介

40 分鐘簡單聊聊 NLP

# What? NLP 簡介

自然語言處理  
(**N**atural **L**anguage **P**rocessing, **NLP**)

= 電腦科學 + 語言學

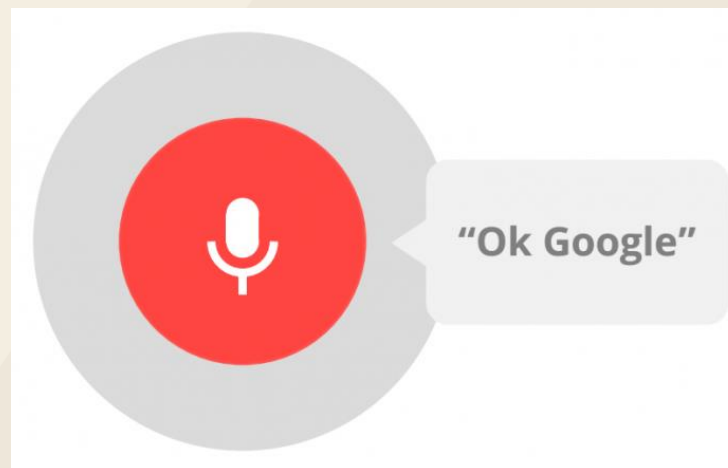
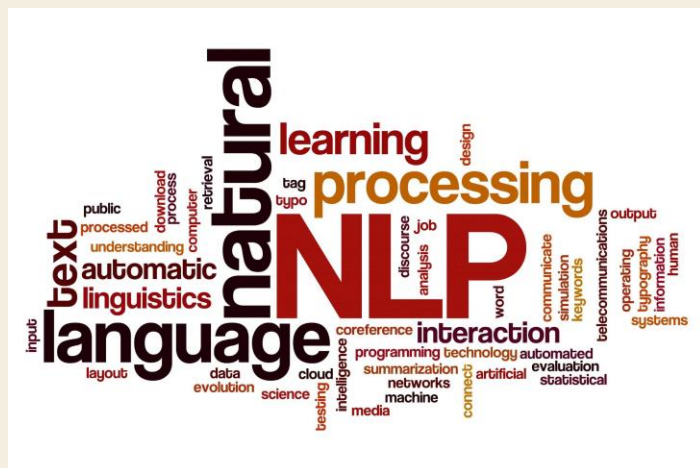
# What? NLP 簡介

探討如何讓電腦理解、運用**自然語言**。

# What? NLP 簡介

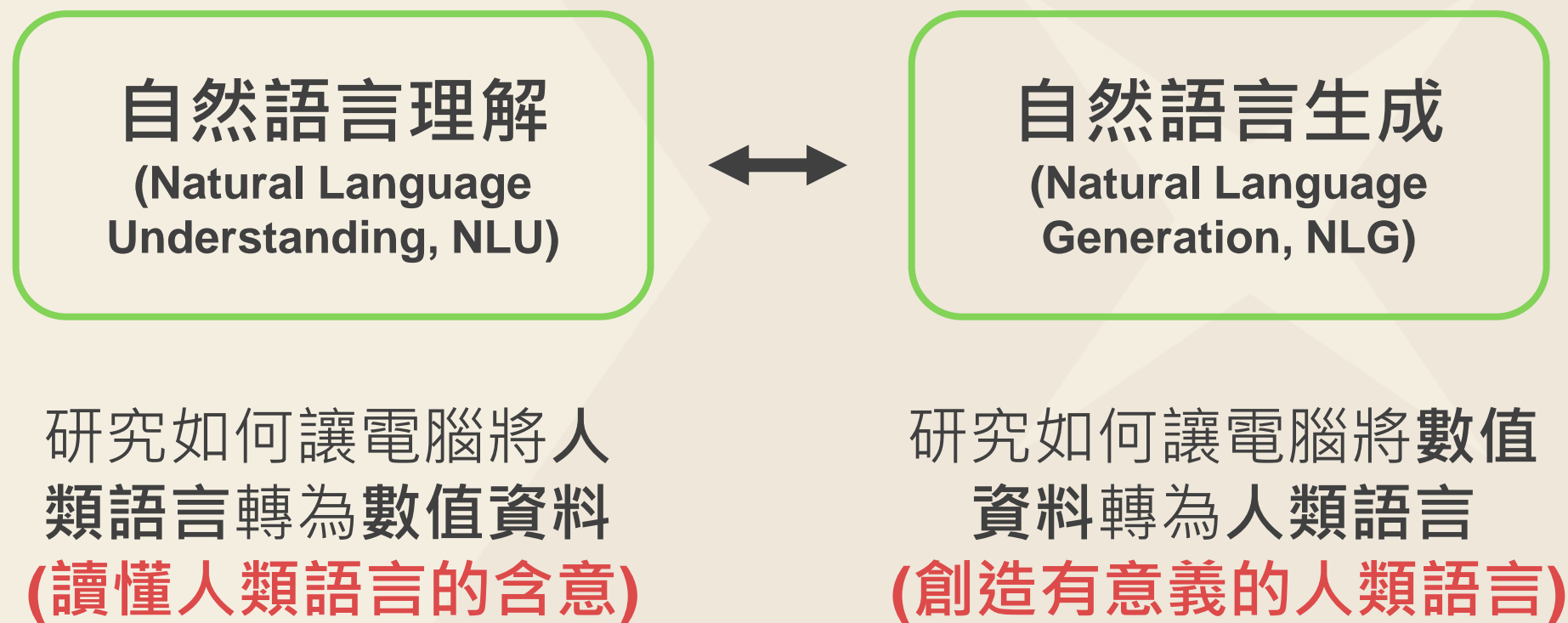
Q：什麼是自然語言？

A：人類為了溝通所創造的語言，通常有特定的文法。形式可以是文字、語音、符號...





# What? NLP 簡介



2

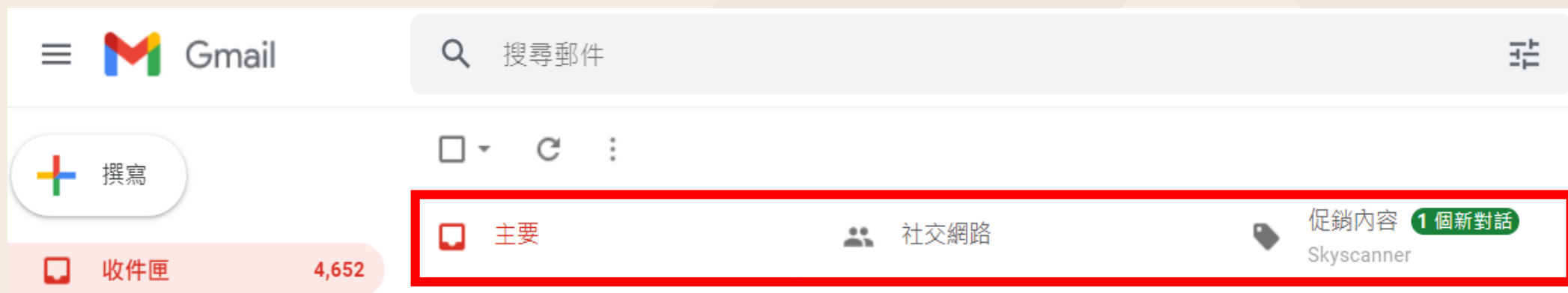
# Why? NLP 實際應用

40 分鐘簡單聊聊 NLP

# Why? NLP 實際應用

## Email 篩選器

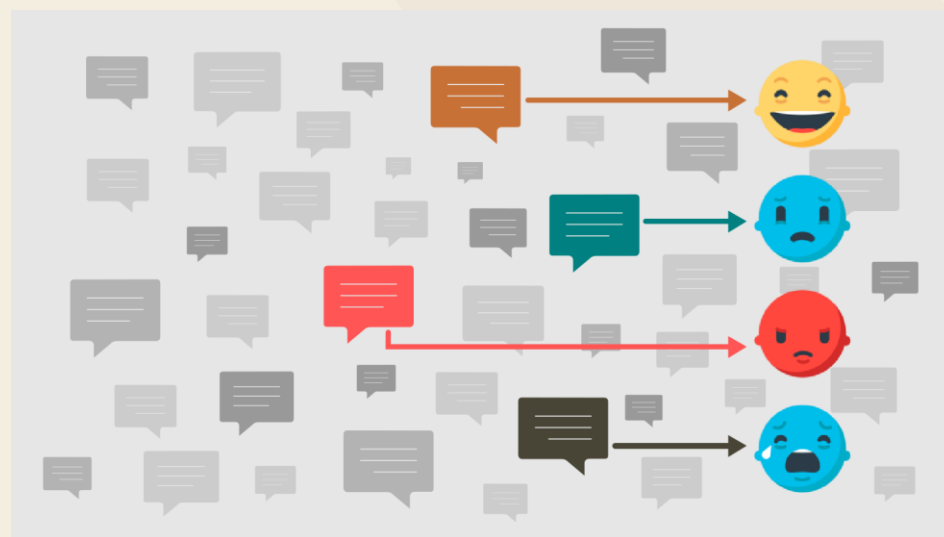
透過信件內容過濾垃圾郵件，或者像 Gmail 將信件分成**主要**、**社交**和**促銷**三類，讓你的收件夾比較整齊。



# Why? NLP 實際應用

## 情感分析

公司可以從社群媒體上蒐集客戶對該產品的相關留言或貼文，分析出**正面**或**負面**的比例，及時改善行銷手法。



# Why? NLP 實際應用

## 智能助理

像是 **Google Assistant**、**Apple Siri**、**Amazon Alexa** 等等，這些智能助理已經漸漸成為日常生活的一部份了，幫助我們處理各種生活瑣事，無聊也可以跟他聊聊天、講笑話~



3

# When? NLP 發展史

40 分鐘簡單聊聊 NLP

# When? NLP 發展史

## 人工規則

在 **1950 年代**，當時還沒有機器學習的概念，因此當時只能透過**語言學分析**語言的規則後，再寫成**電腦程式**。

想也知道，這種方法一定很差，因為語言有太多例外了，**有時連人類都搞不太清楚了，更何況是電腦呢？**

# When? NLP 發展史

到了 **1980 年代**，NLP 開始代入**機器學習**的概念。  
做法是先收集非常多的文本，稱之為**語料庫**。  
然後**訓練模型**從中找出文本中單字間的關聯。  
這比起人工規則更有彈性，也較能處理超出範圍的問題。  
介紹幾個比較有名的模型：



# When? NLP 發展史

## Word2Vec

**Word2Vec** 的輸入是單字，輸出就是代表此單字的向量。

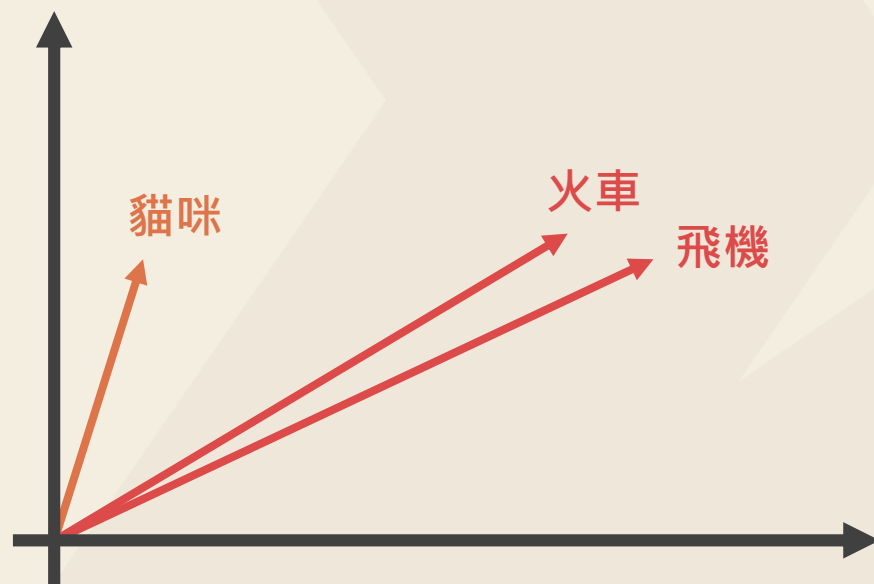
訓練過程簡單講就是透過上下文來學習單字間的關係，比如說：

我明天要搭火車去台北

我明天要搭飛機去台北

# When? NLP 發展史

上頁例子中，火車和飛機的**前後文一致**，因此當模型讀過很多這樣的句子之後，就會給火車和飛機相近的向量。



# When? NLP 發展史

## RNN

**Word2Vec** 看似很棒，但還是有些缺點：

1. 無法處理一字多義
2. 不考慮詞的先後順序

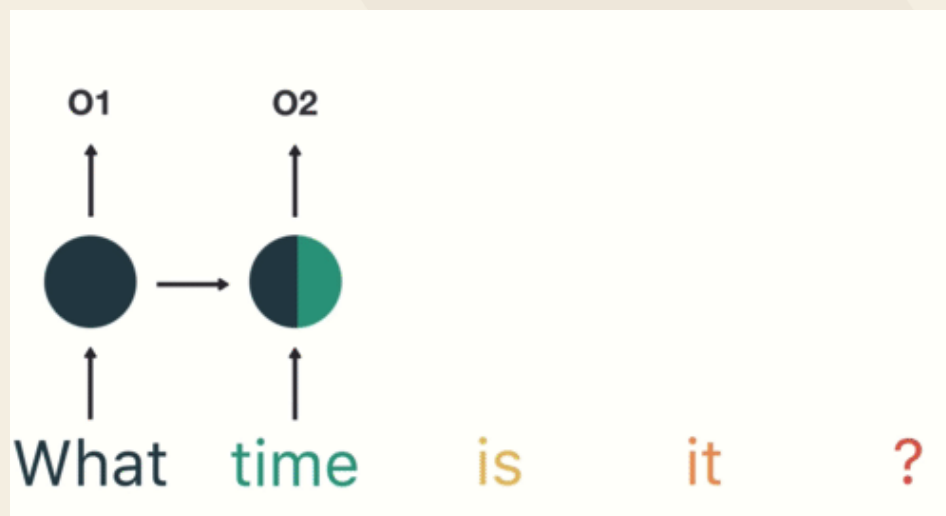
舉例：

我借小明100元

小明借我100元

# When? NLP 發展史

**RNN** 就像我們在閱讀時，不會只看其中一個字，而是從左開始一個一個字讀過來。如此一來就可以根據前文而產生不同的結果。



# When? NLP 發展史

# 語言預訓練模型

自從 2018 年 Google 提出 **BERT** 語言預訓練模型後，對 NLP 帶來革命性的突破。



## 40 分鐘簡單聊聊 NLP

# When? NLP 發展史

Google 收集了大量的語料庫 (Books Corpus + English Wikipedia 總共 33 億個字)，透過**非監督**的方式來 **pre-train**。

之後拿這個 pre-train 好的模型，針對特定的下游任務作 **fine-tune**，結果都比之前的模型還好！

當年橫掃了很多 NLP 任務的排行榜。

# When? NLP 發展史

這概念其實就像我們學中文，從小到大都在接觸中文，耳濡目染之下就有**基本的語感**。

如果此時再叫我們去學**特定的任務** (如：接龍、照樣造句...)，比起從未接觸過中文的外國人，對我們而言就輕鬆很多，對吧？

**pre-train** >> 學會基本語感

**fine-tune** >> 學習特定的任務

# When? NLP 發展史

但除非是大企業，不然要自己從頭開始 pre-train 語言預訓練模型根本是天方夜譚。

做為參考，訓練一個 **1.1億** 參數的 **BERT-BASE** 模型，**要用 16 個 TPU 跑 4 天!** 何況還要先收集那麼大的語料庫。

幸好 BERT 作者有開源 pre-train 好的模型，讓我們可以直接站在巨人的肩膀上，讓下游任務變得既輕鬆又有效。





**NLP研究者**

**BERT**

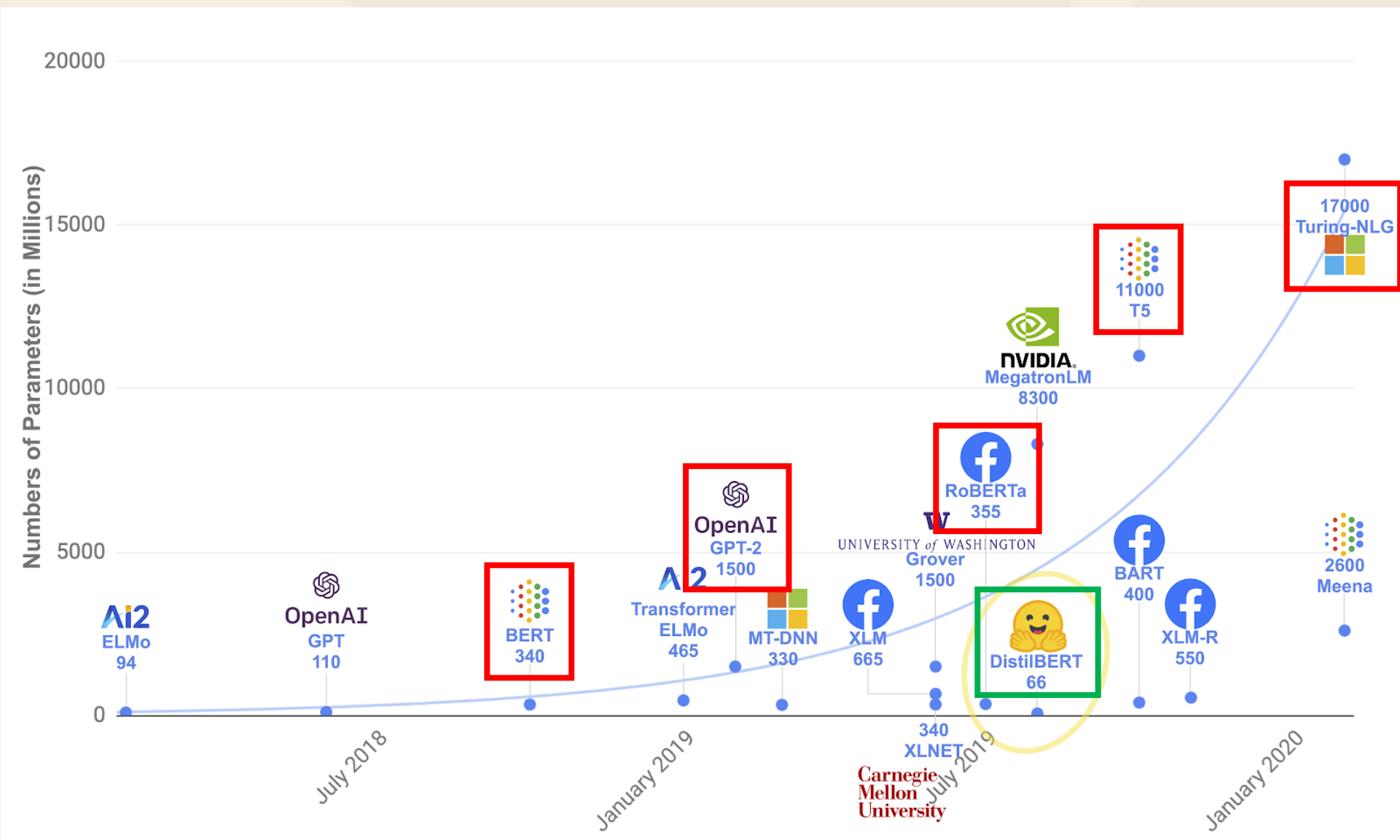
**傳統模型**

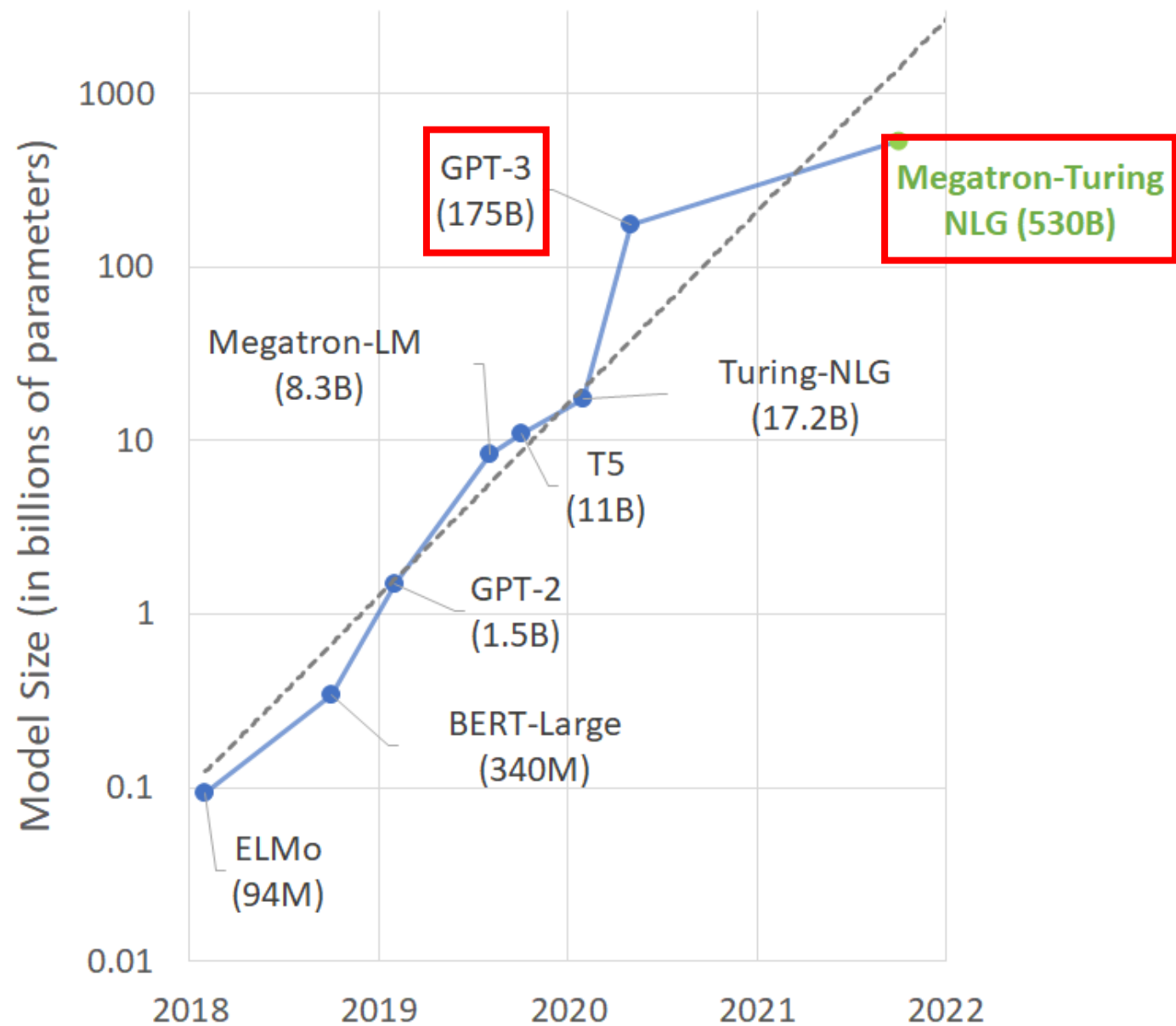
# When? NLP 發展史

Google 提出了 BERT 後，可想而知，其他大企業或組織也競相推出了自己的語言預訓練模型。

後面就是模型**參數一個比一個大**，整個就很扯。

當然也有些模型專注在相同效果下**減少參數**。





4

**How?**

**Google Colab + Hugging Face 實作**

40 分鐘簡單聊聊 NLP



# How? Google Colab + Hugging Face 實作

首先介紹 **Google Colab**，使用過 **Jupyter notebook** 的人，相信對 Colab 就一定不陌生，下面列出一些優缺點：



# How? Google Colab + Hugging Face 實作

優點：

- 不需要架設環境，只要有網路和瀏覽器就可以執行 Python 程式
- 原本就內建許多機器學習的套件
- 免費使用 **GPU**、**TPU**
- 因為存放在 Google Drive 上，所以不管要分享還是共用都很容易
- 視覺化呈現執行結果 (圖片、表格...)

# How? Google Colab + Hugging Face 實作

缺點：

- 連續運行時間最長為 **12 小時**，超過就會被強制停止，而且重啟資料會被清除
- **GPU、TPU 有用量限制**

雖然有缺點，但整體來說還是利大於弊。因此很推薦機器學習的初學者使用！



# How? Google Colab + Hugging Face 實作

**Hugging Face** 是一間人工智慧的新創公司。

開源很多 NLP 領域知名的**語言預訓練模型** (如 BERT、GPT-2...)，支援 **100 多種語言** 的文本分類、文本生成、問答等任務。



## Hugging Face

# How? Google Colab + Hugging Face 實作

其下的 **Transformers** 套件，使用者可以輕易地下載、訓練、上傳語言預訓練模型。


此套件目前在 GitHub 上已經有 **6.7 萬個 star**，成長速度是新創公司中史上最快的。

# How? Google Colab + Hugging Face 實作

接下來就示範怎麼使用 Google Colab + Hugging Face 來實作一些簡單的 NLP 任務吧!

[Colab連結](#)

# Reference

- 斷開中文的鎖鍊！自然語言處理 (NLP)是什麼？
- NLP自然語言處理 – 技術原理與其產業應用
- 進入 NLP 世界的最佳橋樑：寫給所有人的自然語言處理與深度學習入門指南
- 進擊的 BERT：NLP 界的巨人之力與遷移學習
- 台大李宏毅教授 - ELMO, BERT, GPT
-  Transformers Document

# Summary

今天介紹了 **NLP** 的實際應用、發展史和 **Google Colab + Hugging Face** 的實作，但這不過是 NLP 的冰山一角而已，還有很多東西沒時間講。

歡迎對 **NLP** 有興趣的人自行研究，也歡迎找我一起討論，大家一起共創良好的社群環境!

本議程響應開源風氣，所以**投影片**和**範例程式碼**都公開在 **GitHub** 上，請自由使用。

# Summary

這是我的個人網頁，有什麼問題歡迎來聯絡我~  
今天的分享就到這邊，謝謝大家!

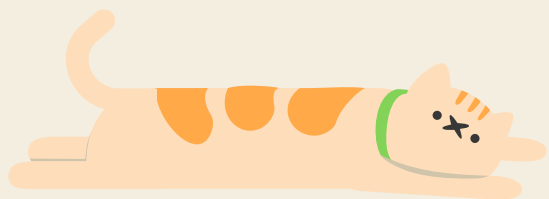


議程投影片+範例程式碼



個人網頁

# Q & A



40 分鐘簡單聊聊 NLP

# Thanks for watching!

