工業工程與管理系

**111 學年度第二學期四技專題製作**

**文章生成**

**專題生: 工管四丙 莊博棱、張家豪**

**指導教授：鄭辰仰 教授**

**中華民國一百一十一年十二月**

# 摘要

近年來，隨著科技的不斷進步，人們生活的步調愈來愈快，又因為少子化的問題導致每個人的工作量愈來愈大，而科技的進步使得人們的個人生產力增高，有著電腦的輔助，繁雜重複的工作逐漸在電腦上形成一個公式，只要簡單的幾個步驟電腦便會將其整理分類完成，統計蒐集數據更是方便，從網路上便可以輕鬆了解現今人們喜好想法。

本次研究便是文章生成，從簡單的文章生成我們便能輕易了解文章內部，加速整理分類等資料處理。

目錄

[摘要 II](#_Toc124007211)

[第一章 緒論 1](#_Toc124007212)

[**1.1** **研究背景與動機** 1](#_Toc124007213)

[**1.2** **研究目的** 1](#_Toc124007214)

[**1.3** **研究流程** 1](#_Toc124007215)

[第二章 文獻探討 2](#_Toc124007216)

[2.1 前言 2](#_Toc124007217)

[2.2 Dataset 2](#_Toc124007218)

[2.3 Tw-rage 2](#_Toc124007219)

[2.4 mt5 4](#_Toc124007220)

[2.5 Train and predict 5](#_Toc124007221)

[第三章 研究方法 7](#_Toc124007222)

[3.1 研究流程 7](#_Toc124007223)

[3.2 環境建置 8](#_Toc124007224)

[3.3 dataset訓練模型並預測 8](#_Toc124007225)

[3.4 資料蒐集 9](#_Toc124007226)

[3.5 資料轉換 9](#_Toc124007227)

[第四章 研究結果 11](#_Toc124007228)

[第五章 結論 13](#_Toc124007229)

[參考資料 13](#_Toc124007230)

# 緒論

* 1. **研究背景與動機**

現代人因為網路大數據等科技進步，使得人們的每天接收大量的資訊，也因此人們閱讀的耐心就越短，譬如youtube與tiktok就可以看出，現今社會較流行段影片，將要訴說的故事濃縮到5分鐘內，也意味著這時代的人大多專注力太過分散，因此現今影音等，大多以短時間大資訊的方式呈現，太多鋪墊或時常過長，便會使人間焦點轉移至其他地方。

* 1. **研究目的**

本研究中，而我們藉由深度學習，使其能夠閱讀文章，並依據文章之前後文，分析出此文章之重點，並且將其濃縮成簡單的一句話，做資料整理也能以十分迅速的型式完成，在工作時，同時有好幾份報告，假使能以文章摘要，那豈不是一眼就可以找到自己所需的報告，以這簡單的方式就可以提高工作效率及便利性。

* 1. **研究流程**

以下為本專題的研究流程。第一章首先說明研究動機與目的；第二章介紹transformers, dataset, tw-rouge, mt5的使用；第三章敘述本專題透過程式所獲取的資料、數據整理、資料預處理及模型訓練的實驗設計；第四章分析與評估實驗結果；第五章針對研究結果進行總結。

# 文獻探討

## 前言

如今運用神經網路進行自然語言的學習並應用已是非常普及，我們根據 github上可以搜尋到的文獻資料進行參考，而這是一個使用 TensorFlow 和 transformers 庫訓練序列到序列 (Seq2Seq) 模型的 python 程式碼。而這篇研究主要就是用來利用新聞內容作為樣本及新聞標題作為標籤，訓練後並用以預測文章標題的標題生成器。

## Dataset

文獻中使用的detaset 是從 udn.com 抓取的，內容包含新聞發布日期(data\_publish)、標題(title)、新聞文章(maintext)及上述所提到的新聞網址(source\_domain)，而後將資料拆分成Train：2015-03-02至2021-01-13共21710篇文章及Public：2021-01-14至2021-04-10共5494篇文章，提供訓練及預測之用，並將使用 jsonlines 格式寫入及讀取資料。

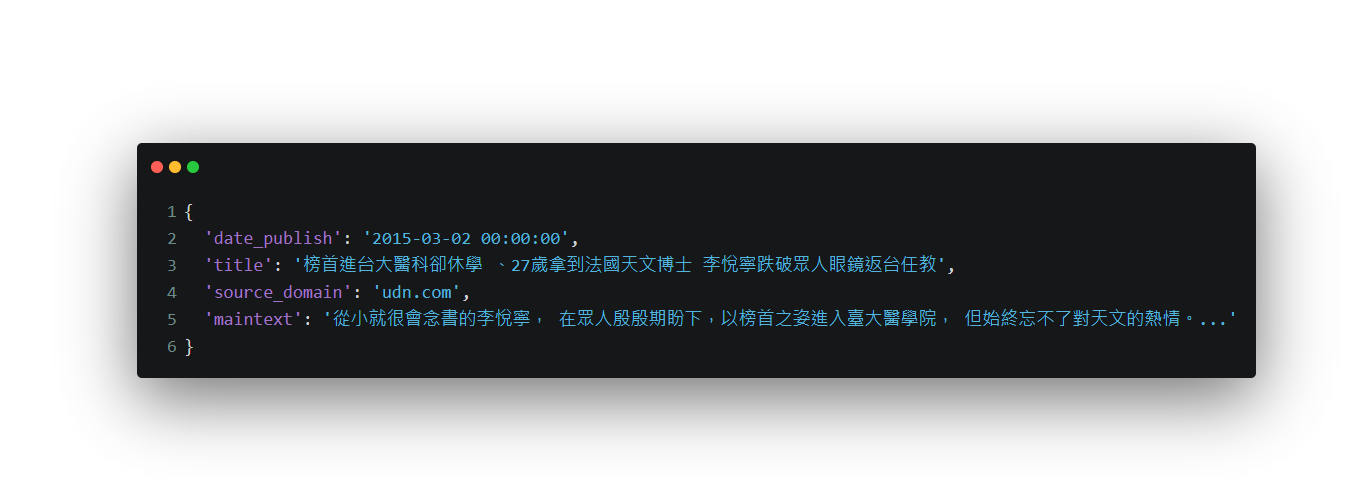


圖2.1

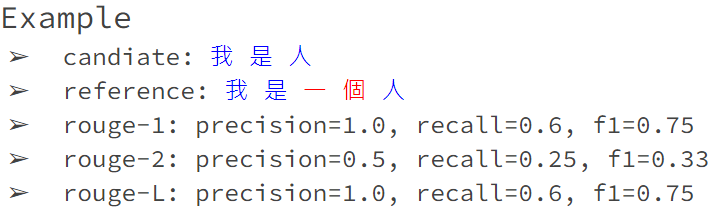
## Tw-rage

文獻代碼中還從另一個Github 網址中下在了Tw-rage 檔案，作用是使用 CKIP 斷詞器和 Rouge 語言指標計算器來處理自然語言處理任務。斷詞是自然語言處理中的一項基本任務，它將一個自然語言句子拆分為單詞，代碼中的 WS 類會使用 CKIP 斷詞器對句子進行斷詞，以便進一步處理。

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) 是一種用於評估文本生成模型性能的語言指標。它計算生成的文本 (generated text) 和參考文本 (reference text) 之間的重疊度，並將其轉換為數值，以便評估生成的文本與參考文本之間的相似度。ROUGE 而這裡使用了三種常用的指標，分別為 ROUGE-1、ROUGE-2 和 ROUGE-L。

* ROUGE-1 是計算生成文本中與參考文本中共有的單詞數量的比例。它會忽略生成文本中單詞的順序，只計算單詞出現的次數。
* ROUGE-2 是計算生成文本中與參考文本中共有的 bigrams (連續的兩個單詞) 數量的比例。它會忽略生成文本中 bigrams 的順序，只計算 bigrams 出現的次數。
* ROUGE-L 是計算生成文本中與參考文本中共有的最長連續子序列的長度的比例。它會考慮單詞的順序，但忽略單詞出現的次數。

圖2.2



ROUGE 分數越高，代表生成的文本和參考文本之間的相似度越高。 ROUGE 常被用來評估摘要生成模型的性能。

在文獻中，get\_rouge 函數接受參考文本和生成的文本，並返回 Rouge 指標的數值。代碼中的 tokenize\_and\_join 函數則會使用 CKIP 斷詞器對文本進行斷詞，然後使用空格將斷詞結果連接在一起。這是因為 Rouge 指標計算器只能處理已斷詞的文本

## mt5

文獻中使用的 mt5-small 從[Hugging Face](https://huggingface.co/) 上下載。該模型訓練資料來自mC4，因此模型具備大規模數據、超過100種跨國語言的理解能力，同時也是一個Text-to-Text的Transformer Model，因此可以應用在多個自然語言理解任務(NLU)，是一個比BERT更為強大與成熟的新方法。它是一個序列到序列（sequence-to-sequence）模型， MT-5 的基本原理是利用自注意力機制（self-attention mechanism）來學習將輸入序列轉換為輸出序列的映射。自注意力機制允許模型在處理每個輸入位置時，將其他輸入位置的信息考慮在內。通過多層自注意力和全連接層的組合，MT-5 可以捕獲輸入序列中的複雜關係和依賴關係，並將這些信息用於生成有意義的輸出序列。

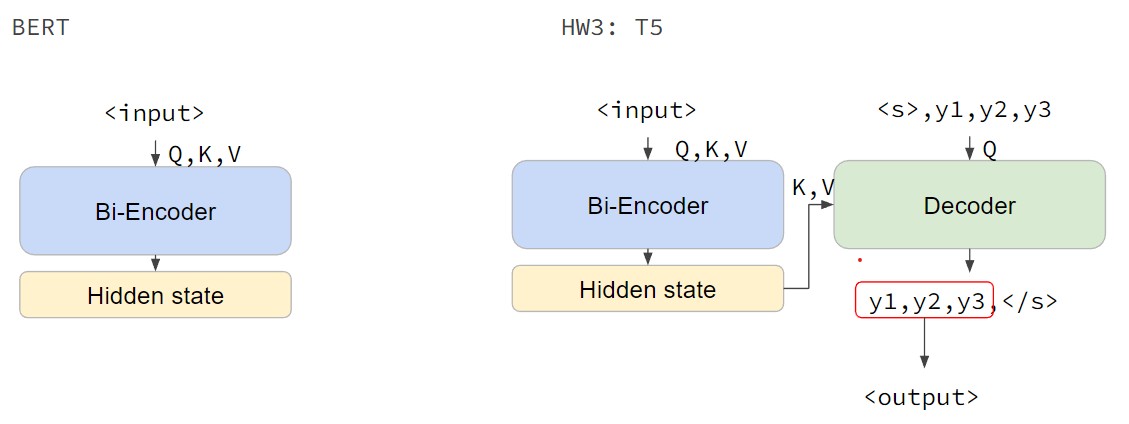
除了自注意力機制之外，MT-5 還使用了其他一些技術來提高性能。例如，它使用了殘差連接（residual connections）和層正規化（layer normalization）等技術，以平滑訓練過程並減少過擬合（overfitting）的風險。此外，MT-5 也支持使用低精度浮點數（FP16）加速訓練。

圖2.2

MT-5 支持超過 100 種語言，並且可以在許多任務上達到最先進的水平。例如，在 WMT 2021 的英語到德語和英語到法語的翻譯評估中，MT-5 分別在新聞和對話類任務上達到了最佳成績。此外，MT-5 還可以用於生成高質量的文本摘要、生成有意義的回答等任務。

MT-5 是一個非常強大的模型，可用於自然語言處理任務，如翻譯、摘要、問答等，但同時也需要大量的訓練資料和計算資源。

## Train and predict

在他給的檔案中，有train.py作為訓練檔。首先會設定 Tensorflow 的輸出等級，然後載入所需的套件和函式。接著會定義一個名為 main() 的函式，並在裡面實現訓練流程。在 main() 函式中，首先會使用 parse\_arguments() 函式解析命令列參數，然後將參數儲存到 args 變數中。接著會使用 set\_seed() 函式設定隨機數種子，並使用 load\_model\_and\_tokenizer() 函式載入models/mt5-small。然後會使用 load\_datasets() 函式載入訓練和驗證資料集。

接下來會使用 Seq2SeqTrainingArguments 建立訓練參數，並使用 Seq2SeqTrainer 建立訓練器，然後使用 trainer.train() 方法開始訓練。最後，程式會定義一個名為 parse\_arguments() 的函式，用於解析命令列參數。

接著執行predict.py用以生成測試文本的標題。代碼會加載mt5-small作為模型和分詞器，並加載要生成文本的測試檔。接下來，代碼會創建數據加載器，並使用訓練迴圈來遍歷數據集。在每個迴圈中，代碼會將輸入文本轉為模型可以處理的張量，並使用模型的 generate 方法生成文本。最後，生成的文本會被使用分詞器解碼並加入預測列表中。另外，代碼中還有一些參數可以控制生成文本的過程，例如 gen\_kwargs 參數可以用來傳遞控制生成文本的選項，如生成文本的最小和最大長度。最後，代碼會將預測結果保存到指定的輸出路徑中。

# 研究方法

## 研究流程

本研究利用github上擁有的深度學習模型，將其轉至與本環境相符之設定，以檔案內部的dataset進行訓練及預測加以驗證程式正確。而後我們透github上的公開資料取得的新聞內容以及新聞標題當作文章摘要，並以此進行模型訓練並加以預測，以新聞標題用以驗證學習正確率，最後再以此資料加以修正模型的正確率。

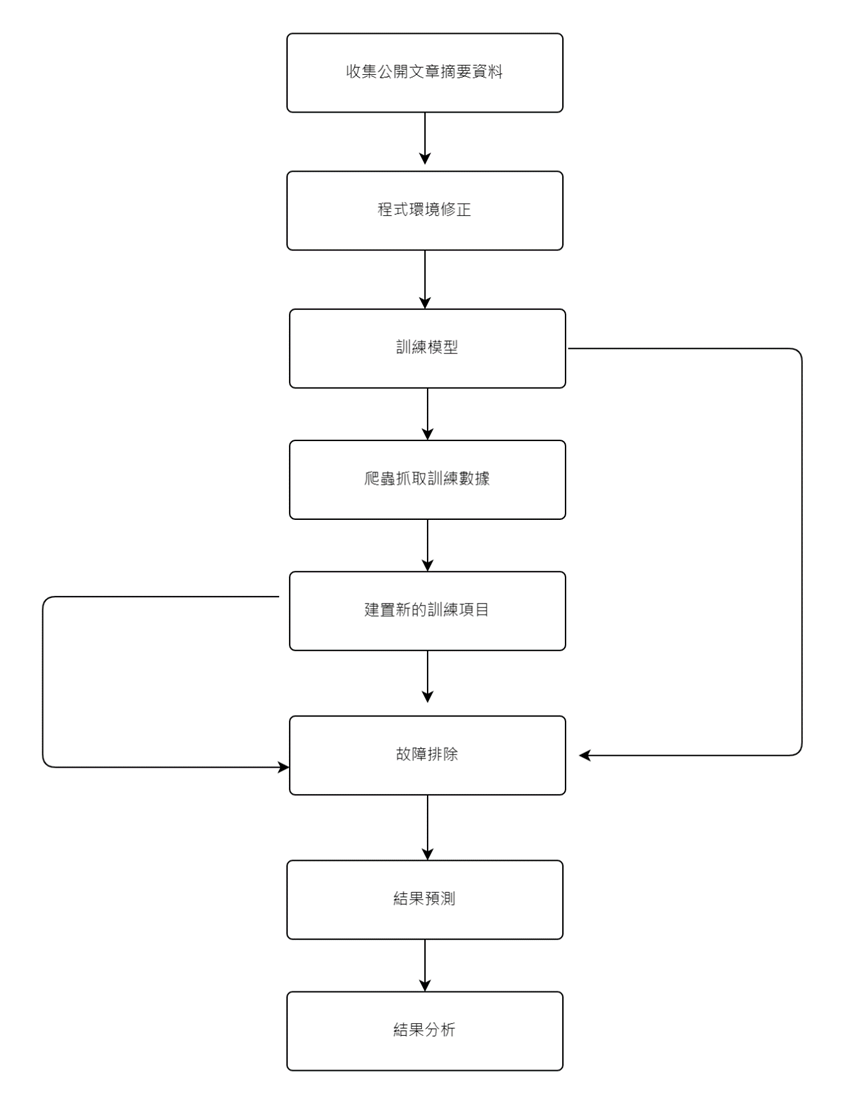
****

圖3.1

## 環境建置

為使程式可以正常運作，我們依照文獻中提供的建議將python環境調整至python 3.9 ，除了安裝了文獻中所指示的[tw\_rouge](https://github.com/ntu-adl-ta/ADL21-HW3/tree/master/tw_rouge" \o "tw_rouge)，也安裝了torch 1.7.1、transformers、sentencepiece等各所需套件，torch1.7.1因執行時須對照CUDA的版本運行，所以我們對應電腦的CUDA版本並以pytorch官網指定的方式安裝，以協助gpu運算加速運算時間。

## dataset訓練模型並預測

我們接著嘗試透過github此文獻檔案中所提供的dataset的Train檔進行訓練接著用Public檔預測來確認模型是否正常運作。在訓練的依開始發生了時間過長的問題，因此我們選擇將程式中 disable\_tensorflow\_gpu() 給去掉，使其能使用gpu去運作。

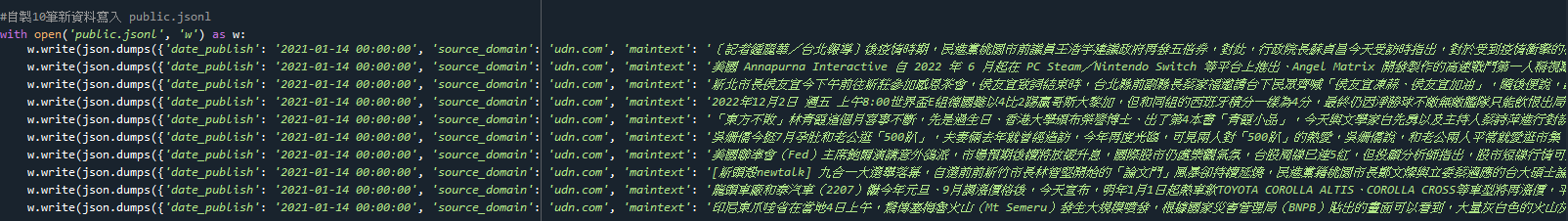
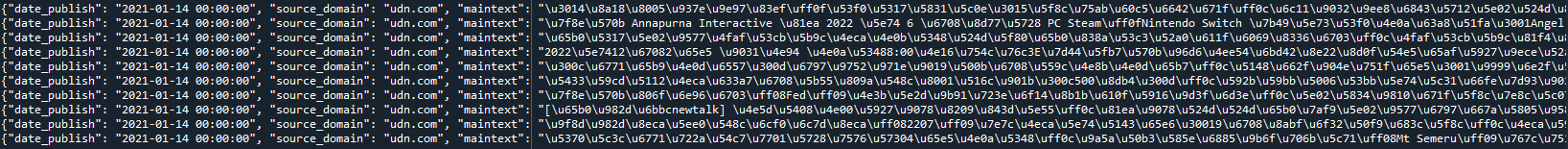
而在訓練完成後我們進行模型第一次預測的測試，使用檔案中所提供的資料，輸出其透過新聞內文所自動生成的標題，之後再試著將自行製作的資料透過jsonlines轉換至與檔案格式相同並進行預測測試，轉和資料如下圖所示。

圖3.2



## 資料蒐集

為了測試其訓練不同資料是否能對預測進行改善或有其他影響，我們從github上下載有共有50000筆包含新聞標題及文章內文等所需數據的資料，如下圖所示。

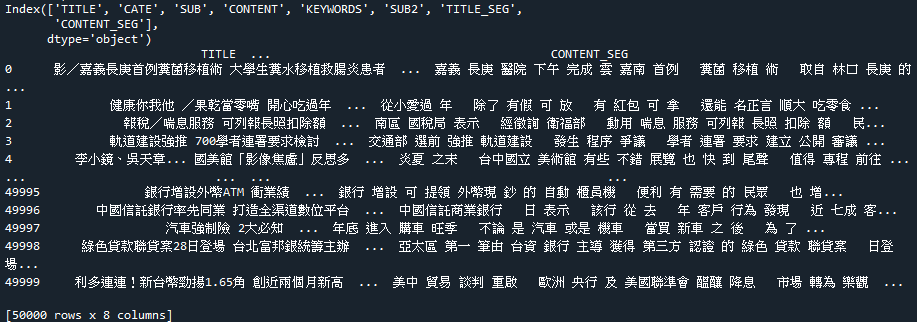
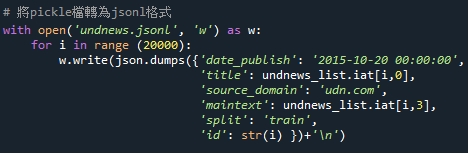


圖3.3

## 資料轉換

我們隨後取資料中的前20000筆作為訓練資料，並將新聞標題及完整內文提取出來，再將中文的資料以askii碼的型式寫入，以便深度學習模型運作，並將其整理成符合輸入的格式，再將檔案轉成jsonl格式放入train.jsonl檔案中，如下圖所示。

圖3.4



隨後我們再將上述20000筆資料進行訓練後接著進行預測，並將其與其他預測進行對比，查看其結果是否有改善及其他狀況。

# 研究結果

**使用原始訓練資料訓練並預測**

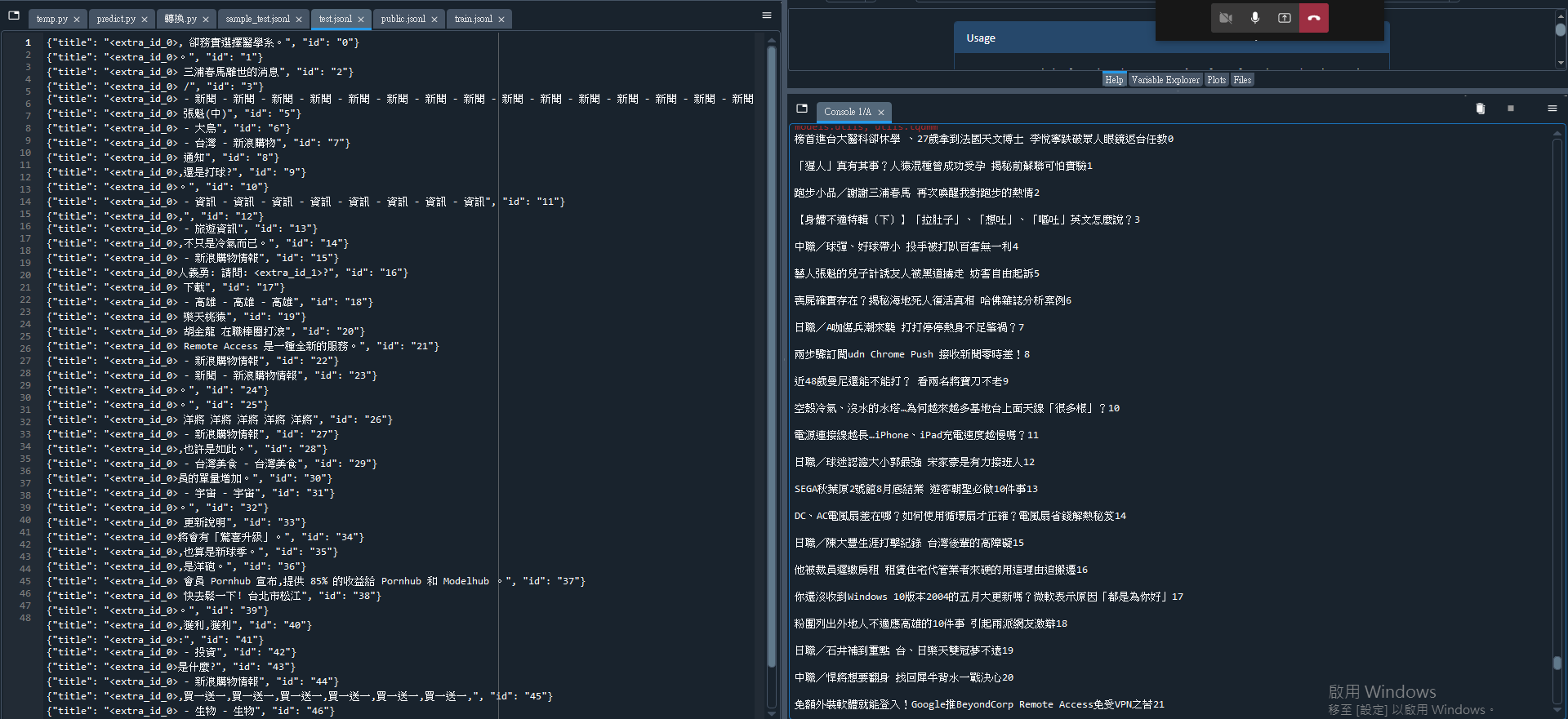
使用檔案中提供的訓練資料完成第一次訓練後，我們首先使用檔案中所提供的Public檔中的5494篇新聞文章進行預測，我們與其新聞盡興對照大多是有偏差的，並且有部分是無意義及空白的預測，但也有少部分能看出文章所表達的部分內容，結果如下圖所示。

圖4.1

而後我們嘗試將自行製作的預測資料進行預測，嘗試其泛用性。過程中我們發現如果只使用5筆資料以下將無法預測，但一開始無法預測的資料部分可能會隨著增加預測資料數變得可以預測，因此我們將資料上升到10筆並進行預測，有部分能預測出結果，但結果依然與實際有一定的偏差，結果如下圖所示。

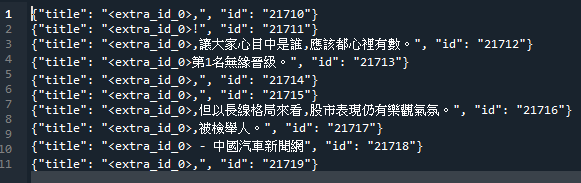


圖4.2

**使用自製訓練資料訓練並預測**

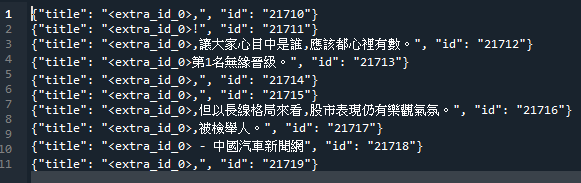
經過上面的測試後我們決定嘗試使用自行準備的數據進行訓練並看看是否有改善及其他影響。為了對照預測結果預先準備10筆預測用資料，並將其先使用上述原始模型進行預測，結果如下圖所示。

圖4.3

接著使用我們自行準備的訓練資料進行第二次訓練，而後使用同樣的10筆預測資料進行預測，結果如下圖所示。

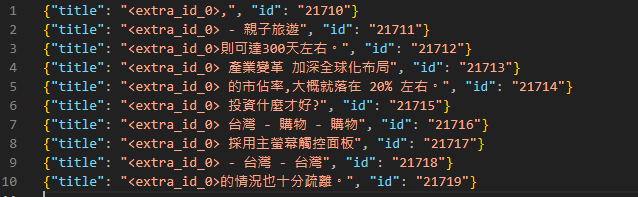


圖4.4

比較後兩者預測結果可以說幾乎是一模一樣，但相較之下多了一筆id:21715資料成功預測，算是有稍微變好了。

# 結論

生成式文本摘要以一種更接近於人的方式生成摘要，這就要求生成式模型有更強的表徵、理解、生成文本的能力。傳統方法很難實現這些能力，而近幾年來快速發展的深度神經網絡因其強大的表徵（representation）能力，提供了更多的可能性，在圖像分類、機器翻譯等領域不斷推進機器智能的極限。借助深度神經網絡，生成式自動文本摘要也有了令人矚目的發展。

而生成摘要與實際新聞標題之差距，我們認為有以下幾點可能

1. 深度學習模型訓練的過程，其所學習之數據不足。
2. Mt5-small參數過少，而文章摘要需模型理解文章之前後文關係，因此文章摘要所需精度以及參數需要更多，可能要使用mt5-large之模型進行學習。
3. 用新聞標題以及內容作學習，本身就可能導致學習不準確，因為新聞為了吸引他人閱讀，可能會使新聞標題與內文不符導致學習不正確。

我們在這測次研究中主要了解了了基於深度神經網絡的生成式文本摘要，了解了如利用ROUGE計算生成的文本和參考文本之間的重疊度及Transformer Mode的運作原理等。自動文本摘要是目前NLP的熱門研究方向之一，我們也從研究中看見生成文本摘要的可行性，並明白還有更多可以嘗試及精進的地方在，也期望在未來不斷嘗試及研究下能夠真正完善生成摘要這項技術。

# 參考資料

娘口大人,当深度学习遇见自动文本摘要, 取自<https://zhuanlan.zhihu.com/p/30559757>

莫烦, Transformer 将注意力发挥到极致, 檢自https://mofanpy.com/tutorials/machine-learning/nlp/transformer

Wu-Jun Pei, [ADL-HW3](https://github.com/jimpei8989/ADL-HW3), 檢自<https://github.com/jimpei8989/ADL-HW3>

谷歌人工智能,mt5-small, 檢自https://huggingface.co/google/mt5-small

[ckiplab](https://github.com/ckiplab), [ckiptagger](https://github.com/ckiplab/ckiptagger), 檢自<https://github.com/ckiplab/ckiptagger>

Wikipedia, ROUGE (metric), 檢自https://en.wikipedia.org/wiki/ROUGE\_(metric)