臺東大學綠色科學學刊　2022，12(1)，pp.57-74

應用深度學習語文字探勘於新聞泛政治化之識別

陳志全、鄭亦君、吳貞儀*、曾育玲

摘要

大數據資料必須要正確客觀，選出泛政治化的文章或資訊，達到反應事實，且不讓政治滲透進我們的討論，汙染到正確的思維，誤導大眾的觀念。而本研究目的為利用臺灣各大新聞文章作為資料來源，透過主題模型，結合深度學習與文字探勘的方式應用在新聞泛政治化之識別，通過 CNN 卷積神經網路計算的詞向量，來判斷文本是否有泛政治化傾向，用上述的訓練方式重複訓練，微調參數，卷積核的尺寸、學習率，進行每次訓練的紀錄，進行比較，取比較好的模型運用在此研究上。在研究中發現利用 TextCNN 模型架構在雲端及本機端，在迭代大約 5700 時，batch 效果未提升，會提早結束訓練，測試損失值 0.3，可高達 90%以上的準確率，本研究判斷出的結果將有利於了解民眾或當前社會的看法。

關鍵字：主題模型、深度學習、文字探勘、自然語言處理

陳志全，國立臺東大學綠色與資訊科技學士學位學程副教授。Email: ccchen@nttu.edu.tw

鄭亦君，台南應用科技大學國際企業經營系教授。Email: t20042@mail.tut.edu.tw

吳貞儀 (通訊作者)，國立臺東大學綠色與資訊科技學士學位學程學生。Email: 10722106@gm.nttu.edu.tw

曾育玲，國立臺東大學綠色與資訊科技學士學位學程學生。Email: 10722110@gm.nttu.edu.tw

Applying deep learning and text mining to identifying para-political opinions of news

Chih-Chuan Chen & Yi-Chung Cheng & Chen-Yi Wu* & Yu-Ling Tseng

Abstract

When using machine learning for big data analytics to identify para-political opinions, the data must be correct and objective. Para-political articles are collected to reflect the facts, and not let politics infiltrate our discussions, contaminate the correct thinking, or mislead the public. The purpose of this research is to use articles from the major news media in Taiwan as a source of information to identify the para-political opinions from the news. The techniques of Topic Model, deep learning, text mining, and CNN (convolutional neural network) are applied. The articles are transformed to vectors to train a CNN to determine whether an article has a para-political tendency. The experimental results showed that the use of TextCNN model in the cloud and the localhost, in the iterations of about 5700, the learning efficiency is not improved. Therefore, the training can be ended early with the test loss value of 0.3, and the accuracy rate can be reached as high as 90%. The results of this study is conducive to understanding the views of the public and the society.

Keywords: Topic Model, Deep Learning, Text Mining, Para-Political

Chih-Chuan Chen, National Taitung University Interdisciplinary Program of Green and Information Technology, Associate Professor. E-mail: ccchen@nttu.edu.tw

Yi-Chung Cheng, Tainan University of Technology Department of International Business Management, Associate Professor. E-mail: t20042@mail.tut.edu.tw

Chen-Yi Wu (Corresponding Author), National Taitung University Interdisciplinary Program of Green and Information Technology. E-mail: 10722106@gm.nttu.edu.tw

Yu-Ling Tseng, National Taitung University Interdisciplinary Program of Green and Information Technology. E-mail: 10722110@gm.nttu.edu.tw

壹、前言

「泛政治化」，指在討論非政治學領域內容、超出政治學限度，在非必要情況下牽扯到政治相關因素。在政黨政治的國家體制內，泛政治化也指政黨之間，為互相競爭而不論對國家人民是否有利，抵制對方政黨的政策或主張。人類生活若受泛政治化影響，則會造成衝突不斷，原有生活時空範圍與政治因素無關者，加入泛政治化因素後，造成原有價值觀判斷與泛政治化之價值觀發生衝突，可能因泛政治化而限縮民眾原有生活言行舉止選擇，或造成民眾僅依照政治認同行事選擇的偏頗現象。在學校教育系統中，如果教師採用泛政治化的意識形態教學，會影響教學效果導致教學品質下降，如此學生不但無法學好專業科目，也學不到正確的政治學內容，只有學到意識形態，泛政治化的社會也會影響人們在不理解問題本質或不經過討論思考的情況下，就做出非理性決定，如此反而會妨礙原本正常發展的政治制度，也可能造成大部分民眾不想關心政治議題，或是將政治無限上綱，認為只要採取某個意識形態就能解決一切問題，成為迷信權威與盲目崇拜現象。(泛政治化介紹,2020.5.6)

我們生活當中，所接收的資訊成千上萬，大數據資料產生的背景離不開我們的生活，又或者是社群網路的興起，人們通過這些媒體傳播資訊及人們自己所傳播的溝通、交流、互動等行為，這些產生的各種資訊皆會被網際網路記錄下來，隨著物聯網的興起，在任何地方，我們直觀所認為不可能被記錄下來產生資料的任何事物，皆有可能被「資料化」。大數據資料的定義有六個Ｖ，分別是 Volume（容量）、Velocity（速度）、Variety（多樣性）、Veracity（真實性）、Value（價值）和 Ｖalence (連通性)，大數據資料的特質和傳統資料最大的不同是，資料來源非常多元，並且種類相當繁多複雜，可以定義為各種來源的大量非結構化與結構化數據，而且資訊更新的速度非常地快，就會導致資訊量大增，重點來了，若是要運用大數據，就不得不注意數據的真實性。

大數據資料必須要正確客觀，挑選出泛政治化的文章或者是資訊，達到反應事實，並且不要讓政治滲透進我們的討論，汙染到正確的思維，誤導大眾的觀念，甚至是有些人看文章標題就會被帶風向等行為。而本研究的目的就是利用臺灣各大社群網站文章(PTT 批踢踢實業坊、Dcard 等)作為資料來源，透過主題模型（Topic Model），結合深度學習與文字探勘(Text Mining)的方式應用在社群網路泛政治化留言之識別，藉由通過 CNN 卷積神經網路計算出來的詞向量，來判斷文本是否有泛政治化傾向，本研究判斷出來的結果將有利於了解於民眾或是當前社會的看法。本研究的研究流程如圖一所示。首先擬定研究主題，進行相關研究的文獻探討，包含主題模型、深度學習、自然語言處理與文字探勘。研究方法的部分包含：資料前處理、主題模型建置與深度學習。最後，判斷文章之泛政治化傾向，由實驗結果的分析進行討論並產生結論。

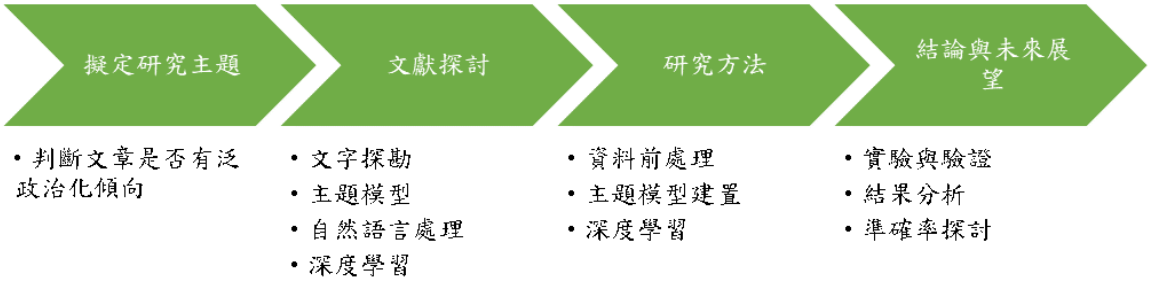


圖 1　研究流程圖

貳、文獻探討

一、自然語言處理

多數自然語言處理系統是以一套複雜、人工訂定的規則為基礎，不過從 1980 年代末期開始，語言處理引進了機器學習的演算法，NLP 產生革新，其成因有兩個：運算能力穩定增加；以及喬姆斯基 語言學理論漸漸喪失主導（例如轉換-生成文法），該理論的架構不傾向於語料庫—機器學習處理語言所用方法的基礎，有些最早期使用的機器學習演算法，例如決策樹，是硬性的、「如果-則」規則組成的系統，類似當時既有的人工訂定的規則，不過詞性標記將隱馬爾可夫模型引入 NLP，並且研究日益聚焦於軟性的、以機率做決定的統計模型，基礎是將輸入資料裡每一個特性賦予代表其份量的數值。許多語音識別現今依賴的快取語言模型即是一種統計模型的例子，這種模型通常足以處理非預期的輸入數據，尤其是輸入有錯誤，並且在整合到包含多個子任務的較大系統時，結果比較可靠。(自然語言處理,2015.10.21)

而自然語言處理的方法對於深度學習的改變，近年來蓬勃發展的深度學習 (Deep Learning)，提出了另一種方法來教電腦表達詞彙。這種方法是將詞彙轉換為「詞向量」，也就是 Word Vector 或稱 Word Embedding，作法是讓電腦閱讀大量文章，利用前後文的統計特性，慢慢學習出每一個詞彙的詞向量，不必利用任何語言學知識。運用「詞向量」的好處是，很多時候針對特定的自然語言處理任務，訓練資料是不足的。因為許多字詞的語義，在人類語感上明明意思很接近、可以相通，但對機器來說，詞彙符號(也就是字元)不同，就是截然不同的詞彙，造成各個詞彙在訓練資料的統計佔比相當低，無法得到足夠信心水準的分析結果。然而，訓練過程中，若我們以「詞彙向量」作統計，在向量空間上，有些字詞間的向量很靠近，團結力量大，就會發現相近的詞彙向量在訓練資料的統計佔比大幅提升，解決了訓練資料不足的困境。同時，詞彙向量在深度學習的模型之中，被視為可修改的參數，所以也具備了語義(詞彙向量)自動調整的能力。(林婷嫻, 斷開中文的鎖鍊！自然語言處理 (NLP))

二、文字探勘（Text Mining）

文字探勘是種透過資訊技術來分析文本的意見、情緒以及感受的技術，在現今資訊爆炸的時代，網路是重要的訊息傳播管道，由於文字探勘能大量且迅速的了解文字訊息中的含意，現今也經常運用於分析報告上，簡單歸納出以下圖一步驟：確立研究

主題、文獻探討、研究方法、實作與驗證、結果分析、結語及未來發展。

文字探勘方面，本研究首重於文章上的資訊擷取，在資料輸入與擷取步驟當中，利用人們通過這些媒體傳播資訊及人們自己所傳播的溝通、交流、互動等行為，這些被網際網路記錄下來產生的各種資訊，記錄下大眾的情緒預測出情緒狀態；進行第二步驟將文字轉為向量並取得文本特徵，定義關鍵字並利用各個字詞出現的頻率次數作為穩本的特徵，在文字探勘時，需經過斷詞處理，將語句分為數個字詞來表示語句的意涵，主要概念是透過機器學習來找到字詞或文本相對應的向量空間，來瞭解與呈現字詞之間的關係，文字呈現的內容大多為非結構化，文字擷取的意義在於將非結構化的資訊變為結構化資訊，以利後續研究的分析與實作。

(一) 斷詞及詞性標註

一般文章中含有大量的詞語與用詞，而詞為文章字句中的最小單位，我們可以將詞句切割，並從中擷取有作用的資訊，了解通篇文章內容及利於往後分析，並可統計出每個詞語的權重。自然處理領域也包含了詞性標註，針對字句中的詞語深入剖析，賦予每個詞合適的詞性。藉由詞性標註，依據各種目的與需求，可以進行特定詞性之詞彙篩選。

(二) 特徵擷取

在機器學習研究詞句的分類領域中，為提升分析之準確率並降低處理文字量，以及對於效能及運算時間也須納入考量，基於此有許多研究學習速度的文章多半聚焦於關鍵特徵值，而非其他無價值之特徵值，將此類詞彙去除也是特徵階段的一項任務。而近年來在我多次查找與對比後，在特徵擷取方法中「TF-IDF(Term Frequency-Inverse Document Frequency)」(Salton & Buckley, 1988) 被廣泛使用。此方法衡量了字詞於文章中的出現頻率(Term Frequency, TF)，以及字詞出現於文集中出現的頻率次數(Document Frequency, DF)，透過上述兩者來衡量關鍵詞於特定文章的重要程度，進而達到詞彙篩選的目的。

圖 2　文字探勘流程圖

三、主題模型建置

主題模型（Topic Model）在機器學習和自然語言處理等領域是用來在一系列文檔中發現抽象主題的一種統計模型。直觀來講，如果一篇文章有一個中心思想，那麼一些特定詞語會更頻繁的出現。一個主題模型試圖用數學框架來體現文檔的這種特點。

主題模型自動分析每個文檔，統計文檔內的詞語，根據統計的信息來斷定當前文檔含有哪些主題，以及每個主題所占的比例各為多少。

主題模型目前已被廣泛運用於機器學習與自然語言處理領域，主題模型主要從文集中找出潛在主題，並透過潛在主題來建構生成模型，本研究主要使用主題模型是潛藏狄利克里分配(Latent Dirichlet Allocation, LDA) (David M. Blei, Ng. Jordan, & Lafferty, 2003)，近年來已經有不少研究將 LDA 運用於自然語言模式、機器學習及圖樣辨識等領域。僅以兩組參數 α 與 β 來代表訓練語料的潛藏語意資訊。首先，假設訓練語料集 D 中共有 M 篇文件，而每一篇文件 d 中有 Nd 個詞，我們先由一組狄利克里分配 α 的參數求得每一篇文件 d 產生所有潛藏主題的機率向量 d，文件中每一個詞在每一個潛藏主題 n，dT 下產生的機率分布則由 β 生成。

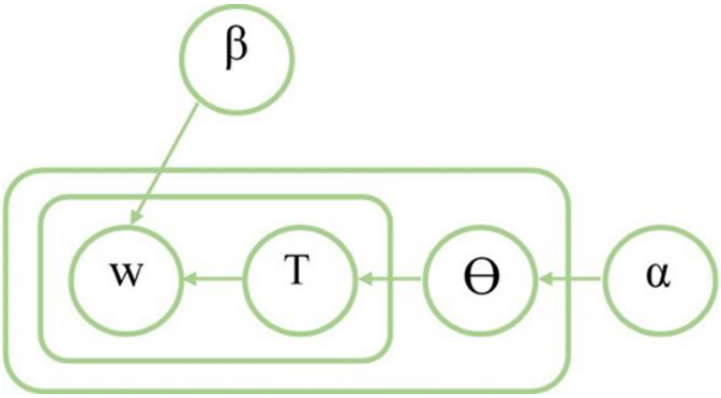


圖 3　LDA 主題模型

四、深度學習

從圖 3 可以看出人工智慧與機器學習和深度學習層層關聯，最早期先有了人工智慧的概念，這也是最廣泛被應用的術語，讓計算機能夠使用邏輯、if-then rules、決策樹和機器學習來模仿人類智能的技術。到了 80 年代出現了機器學習，此技術包含了豐富的統計技術，而機器會根據這些經驗來改善他的任務。最後則在近十年才發展出了深度學習，這是機器學習中一種基於對數據進行表徵學習的算法，透過模仿生物神經系統的數學模型，進行多次運算和訓練，找出最佳化的深度學習模型。和機器學習相比，深度學習只需要在設計好的神經網路中，藉由訓練資料與參數的設定，機器就能自行學習，並演算出最好結果。(全球 AI 人才趨勢,2020.01.14)

深度學習（Deep learning）是機器學習的分支，也是一種以人工神經網路為架構，對資料進行表徵學習的演算法，也是機器學習中一種基於對資料進行表徵學習的演算法。觀測值可以使用多種方式來表示，如每個像素強度值的向量，或者更抽象地表示成一系列邊、特定形狀的區域等，而使用某些特定的表示方法更容易從實例中學習任務（例如，臉部辨識或面部表情辨識），深度學習的好處是用非監督式或半監督式的特徵學習和分層特徵提取高效演算法來替代手工取得特徵。(2021.02.13,深度學習)

隨著智慧科技突飛猛進，讓機器讀懂人類的心情已經不再遙不可及，情感運算產品也朝向多元化發展，有許多業者從情感運算的角度切入，分析聲音、表情、行為，全方位滿足人類情感需求與安全保障，試圖打造有溫度的服務。未來，情感運算勢必會更深入各領域的應用，並持續累積各領域知識，結合人工智慧科技提供更有感的服

務，有意朝情感運算發展業者，應思考跨域整合應用並結合生態系累積更多數據，提供更貼心的服務，以掌握這波情感商機，情感運算在近期機器學習、深度學習等人工智慧技術突飛猛進發展後，已開始展現在多樣的應用上，例如機器人或無人機協助物資運送，透過紅外線熱像儀、臉部辨識技術降低人員接觸感染的可能性，或依據人所在位置、情境乃至於動作，主動的提供個人化、情境化的調適與回應，可見，發展更有認知能力、具備情緒溫度的機器人或智慧助理將是下一波人工智慧的發展方向。(資策會 MIC 資深產業分析師朱師右，2020)

發展至今，深度學習所遭遇的瓶頸大致可分為三類，深度學習需依靠大量數據學習，而所遭遇的瓶頸全都和數據有所關連，深度學習雖然優於其他技術，但並非通用的，經數年發展，瓶頸也逐漸凸顯出來，主要為下列三點：

- 需大量標註的數據
  深度學習能夠實現的前提是大量經過標註數據，雖然有些方法可減少對於數據的依賴，如：少樣本學習、非監督式學習等。
- 過度擬合基準數據
  深度神經網路在基準數據集上表現優異，但在真實世界的圖像上，效果則差強人意，實際的應用當中，如深度網絡有偏差，將會帶來很嚴重的後果。
- 圖像變化過度敏感
  深度神經網絡對標準的對抗性攻擊很敏感，這些攻擊會對圖像造成人類難以察覺的變化，但可能會改變神經網絡對一個物體的認知，對於任何一個目標對象，數據集中只在有限數量的場景，實際的應用中，神經網絡會明顯偏向這些場景。

參、研究方法

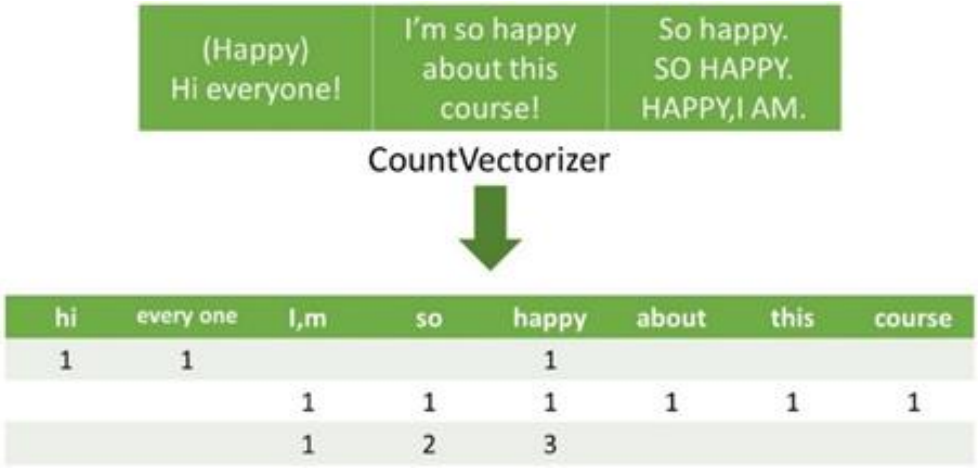
從圖 1 中可看出本研究的研究方法包含：資料前處理、主題模型與深度學習。首先，從臺灣各大社群網站文章(PTT 批踢踢實業坊、Dcard 等)作為資料來源將進行一系列的資訊擷取處理，包括：自然語言分析的處理、斷詞及詞性標註。接著，萃取結構化文字，並用 LDA 萃取主題。藉由通過 CNN 卷積神經網路計算出來的詞向量，應用在社群網路泛政治化留言之識別，本研究判斷出來的結果將有利於了解於民眾或是當前社會的看法。

一、資料前處理

透過資料前處理的相關工作，擷取出情緒化詞彙並存入情緒資料集。此階段細分為三步驟，斷詞與斷句標註、向量維度縮減，以及特徵選取。目前常用中文處理系統有中研院的 CkipTagger 中文處理工具 GPL-3.0（GNU General Public License v3.0）, SnowNLP, 與結巴(Jieba)等。

向量表示法(Bag of Words,BOW)，用字詞向量來表示抓取的文章中的字詞，作為語意資料的依據，每一個字詞都是向量中的其中一個維度，把文字轉化成向量，而向

量中的每一個數值代表該字詞在向量中的重要程度，詞袋模型（Bag-of-words model）是個在自然語言處理和信息檢索(IR)下被簡化的表達模型，此模型下，一段文本可用一個裝著這些詞的袋子來表示，這種表示方式不考慮文法以及詞的順序，詞袋模型被廣泛應用在文件分類,詞出現的頻率可以用來當作訓練分類器的特徵。(曾元顯,2012.10)



hi	every one	I,m	so	happy	about	this	course
1	1			1			
		1	1	1	1	1	1
		1	2	3			

圖 4　4 LDA BOW 詞袋的概念

在特徵擷取方面，本研究使用較為傳統資訊擷取方式，以 TF-IDF 方式是一種用於資訊檢索與文字挖掘的常用加權技術。tf-idf 是一種統計方法，用以評估一字詞對於一個檔案集或一個語料庫中的其中一份檔案的重要程度。字詞的重要性隨著它在檔案中出現的次數成正比增加，但同時會隨著它在語料庫中出現的頻率成反比下降。tf-idf 加權的各種形式常被搜尋引擎應用，作為檔案與使用者查詢之間相關程度的度量或評級。除了 tf-idf 以外，網際網路上的搜尋引擎還會使用基於連結分析的評級方法，以確定檔案在搜尋結果中出現的順序。來計算各個情緒詞於評論中的重要性，並設定門檻值 。當字詞之 TF-IDF 數值低於門檻值時，將此字詞於評論中去除，最後得到較具代表性的情緒詞集。透過篩選後的情緒文集，來呈現各篇評論。(TF-iDF ,2020.12.10)

二、 主題模型建置

本研究主要使用主題模型是潛藏狄利克里分配(Latent Dirichlet Allocation, LDA) (David M. Blei, Ng. Jordan, & Lafferty, 2003)，主題模型目前已被廣泛運用於機器學習與自然語言處理領域，主題模型主要從文集中找出潛在主題，並透過潛在主題來建構生成模型，模型的建置流程可以分為四步驟：資料前處理、模型參數設定、參數最佳化,最後為模型結果的彙整。透過搜集而來的文件資料,經由潛藏狄利克里分配(Latent Dirichlet Allocation, LDA)的處理，分別記錄文件與主題、以及主題與字的關聯性。通過對文章進行“放大”和“縮小”就可以得到較具體或者較粗略的主題；在文件中就可以看到這些主題是如何隨著時間變化，或者說是如何相互聯絡的。搜尋文件就不只是通過關鍵詞尋找,取而代之的是先找到相關的主題,然後再查詢與這一主題相關的文件。(itread01, 2018.12.22)

三、深度學習

本研究進一步使用深度學習進行文章是否為泛政治化傾向之判斷，應用在社群網路泛政治化留言之識別，藉由通過 CNN 卷積神經網路計算出來的詞向量，來判斷文本是否有泛政治化傾向，本研究判斷出來的結果將有利於了解於民眾或是當前社會的看法。

Convolutional Neural Network (CNN) 卷積神經網路，整個 CNN 結構主要分成幾個部分：

■ 卷積層 (Convolution layer)

卷積層主要是由許多不同的 kernel 在輸入圖片上進行卷積運算。Convolution 原理是透過一個指定尺寸的 window，由上而下依序滑動取得圖像中各局部特徵作為下一層的輸入，利用 filter 在輸入圖片上滑動並且持續進行矩陣內積，兩個步驟組成的運算：滑動 + 內積，這個 sliding window 在 CNN 中稱為 Convolution kernel，卷積後得到的圖片我們稱之為 feature map。透過 kernel window 來進行，只是 CNN 利用此方式來取得圖像中各局部的區域加總計算後，透過 ReLU activation function 輸出為特徵值再提供給下一層使用。

■ 池化層 (Pooling layer)

在 Pooling Layer 這邊主要是採用 Max Pooling，Max Pooling 的概念很簡單，只要挑出矩陣當中的最大值，Max Pooling 主要的好處是當圖片整個平移幾個 Pixel 的話對判斷上完全不會造成影響，以及有很好的抗雜訊功能。

■ 全連接層 (Fully Connected layer)

將之前的結果平坦化後接到最基本的神經網絡。

Deep learning 中的 CNN 較傳統的 DNN 多了 Convolutional（卷積）及池化（Pooling）兩層 layer，用以維持形狀資訊並且避免參數大幅增加。在加入此兩層後，我們所看到的架構就如下圖分別有兩層的卷積和池化層，以及一個全連結層（即傳統的 DNN），最後再使用 Softmax activation function 來輸出分類結果。

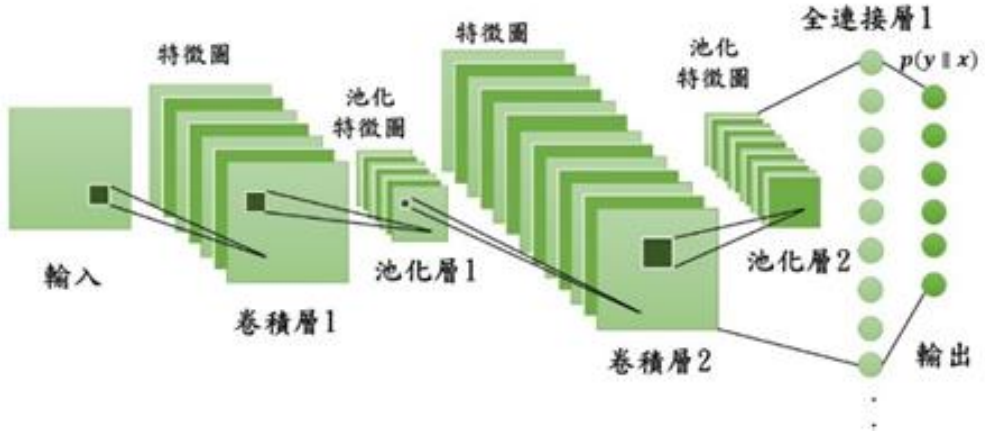


圖 5　CNN 概念圖

肆、結果與討論

從圖 1 中可看出本研究的研究方法包含：資料前處理、主題模型與深度學習。首先，從臺灣各大社群網站文章(PTT 批踢踢實業坊、Dcard 等)作為資料來源將進行一系列的資訊擷取處理，包括：自然語言分析的處理、斷詞及詞性標註。接著，萃取結構化文字，並用 LDA 萃取主題。藉由通過 CNN 卷積神經網路計算出來的詞向量，應用在社群網路泛政治化留言之識別，本研究判斷出來的結果將有利於了解於民眾或是當前社會的看法。

一、環境介紹

本研究使用 Colaboratory，簡稱 Colab，是一個在雲端運行的編輯執行環境，由 Google 提供開發者虛擬機，並支援 Python 程式及機器學習 TensorFlow 演算法。Colab 目的在提供 Machine Learning 機器學習教育訓練及研究用，不須下載、不須安裝就可直接應用 Python 2.7 與 Python 3.6 資源庫。程式碼預設會直接儲存在開發者的 Google Drive 雲端硬碟中，執行時由虛擬機提供強大的運算能力，不會用到本機的資源。使用 Colab 不需自行安裝 TensorFlow 等函式庫，直接 import 即可。提供 None (由 google 配置)、GPU(圖形處理器)及 TPU(張量處理器 Tensor Processing Unit)。

GPU：NVIDIA TESLA V100

二、資料前處理

透過資料前處理的相關工作，擷取出情緒化詞彙並存入情緒資料集。此階段細分為四步驟，以爬蟲收集文本資料、斷詞與斷句標註、向量維度縮減，以及特徵選取。目前常用中文處理系統有中研院的 CkipTagger 中文處理工具 GPL-3.0（GNU General Public License v3.0）, SnowNLP, 與結巴(Jieba)等。

(一) 爬蟲－ET Today 新聞標題及網址

設定目標網址：ET Today 的新聞頁面，獲取 user_agents。

有些網站是不允許被爬蟲的，所以可以設定 UA 假裝自己是瀏覽器，UA 會告訴網站它是透過什麼工具（通過 UA 分析出瀏覽器名稱、瀏覽器版本號、渲染引擎、操作系統）發送請求的，就可以騙過該網站，防止 IP 被鎖。

－使用 request 傳給對方主機時，要附上 headers 資料，讓行為更像瀏覽器

－header 裡面放 user-agent(UA)，每次的 request 都用不同的 UA，使用隨機的 UA

指派隨機選擇出的 list item，在 headers 中，指派一個從上面 list 中隨機選擇出的 list item(一個 user-agent)，就可以開始進行 request。當我們傳一個 request 給主機後，主機（假設沒有其他資安或防爬蟲的檢驗）即會自動回傳一個 response 內容，存在 resp 中（其實主要就是 Html+CSS+JS 等），解析 resp 的內容，把標題、文章網址、日期、分類分別存進 list 中，最後把這四個 list，匯入 Dataframe 物件 df 中。匯出結果至 csv 檔。

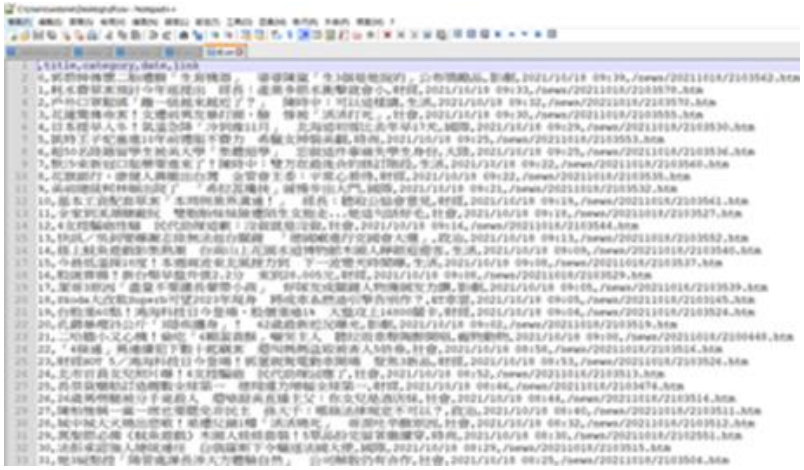


圖 6　驗證集檔案示意圖

(二) 資料整理

從 ETToday 的新聞頁面中抽取了二十萬條新聞標題，並把這二十萬條新聞標題以 1：1：18 切割資料集，儲存成訓練集、驗證集、測試集檔案，驗證集內含共一萬條新聞，測試集內含共一萬條新聞，訓練集內含共十八萬條新聞。

圖 7　測試集檔案示意圖

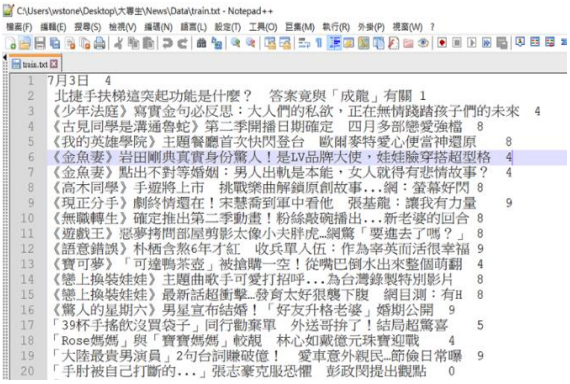


圖 8　訓練集檔案示意圖

並且把全部共二十萬條新聞標題，做類別上的標記，以下共分成 10 個類別。

類別：Sports、Politics、International、Finance、Fashion、Live、Society、3c、Games、Movies。

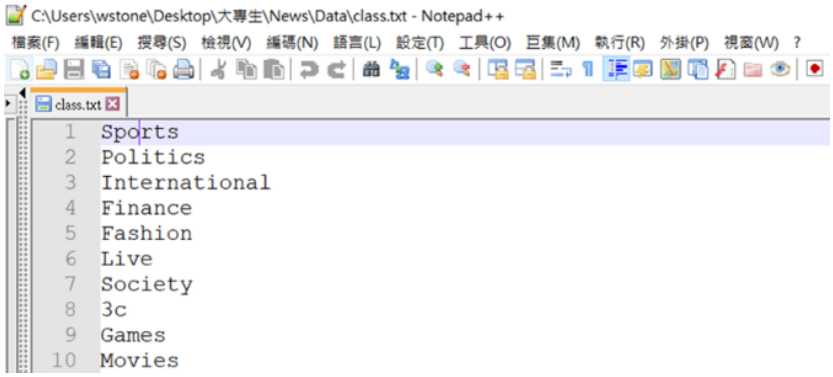


圖 9　類別檔示意圖

三、Ckip

運用到中央研究院開發的中文情感語意分析套件，中文情感語意分析套件(CSentiPackage)包含多個可以用於中文情感語意分析研究所需要的各式工具，包括中文意見與構詞詞典、中文意見樹庫、意見挖掘計分工具，及深度社群立場分析模型，可協助探勘目標文本所表達的意見或情感及其強度，技術優勢在套件包含中文情感與意見分析所需要的所有工具，可解決中文環境中情感與意見分析技術資源不足的發展困境，使大量應用的開發成為可能。套件提供的 UTCNN 模型（古倫維，2019），是一個不限語言皆可使用的模型，可以用在社群媒體與論壇之中，同時也是情感與意見分析領域最具應用性的產品評論及輿情分析兩大議題上，為到目前為止較高的工具。

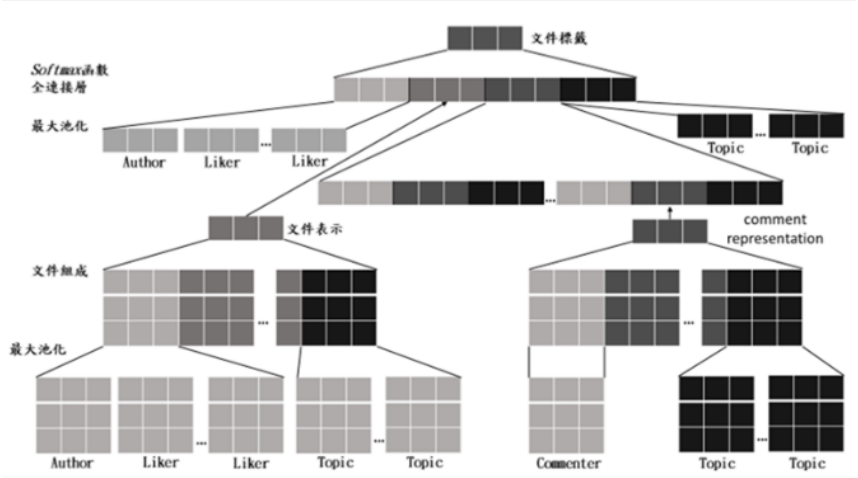


圖 10　UTCNN 模型架構

工具分為三個等級（1—3），等級一最快，等級三（預設值）最精準。

■ (WS) Word Segmentation 斷詞

■ (POS) Part-of-Speech Tagging 詞性標記

■ (NER) Named Entity Recognition 實體辨識

斷詞與實體辨識的輸入必須是 list of sentences，詞性標記的輸入必須是 list of list of words。詞性標記工具會自動用 '，,。：:；;！!？?' 等字元在執行模型前切割句子（輸出的句子會自動接回），可設定 delim_set 參數使用別的字元做切割。

另外可指定 use_delim=False 已停用此功能，或於斷詞、實體辨識時指定 use_delim=False 已啟用此功能。

```
高
高(Nb)
嘉
嘉(Nc)
瑜
瑜(Nb)
NerToken(word='瑜', ner='PERSON', idx=(0, 1))
```

圖 11　CKIP 示意圖

四、Jieba&LDA

(一) Jieba

主要是透過詞典，在對句子進行斷詞的時候，將句子的每個字與詞典中的詞進行配對，找到相符合匹配的詞則斷詞，否則沒找到則無法斷詞。而主要是在於，相連的字詞在不同的文本中出現的次數越多，就推斷這相連的字很可能就是一個詞。

Jieba 是 Python 環境中非常常被用到的中文斷詞工具，其提供三種分詞模式：

- 精確模式：試圖將句子最精確地切開，適合文本分析。
- 全模式：把句子中所有可以成詞的詞語都掃描出來，速度非常快，但是不能解決歧義。
- 搜尋引擎模式：在精確模式的基礎上，對長詞再次切分，提高召回率，適合用於搜尋引擎分詞。

本研究選用 Jieba 斷詞是因為其速度快，如是做簡單的機器人且沒有那麼在乎準確性，此外也可以用在文檔分析上，使斷詞以便之後進行更進一步的操作。

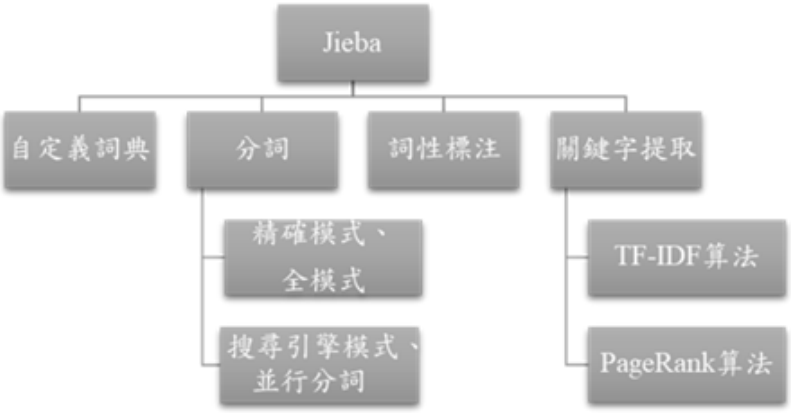


圖 12　Jieba 整體功能圖

(二) LDA（Latent Dirichlet Allocation）

LDA 的基本精神，是先定義好有限的主題，並透過觀察文件與用詞來計算出主題之間的關聯，以及各個文件的主題分佈，只要文件量夠多，就可有效的快速理解不同文件的主題分佈量。

LDA 有兩個基本的原則：

- 每篇文件都是由數個「主題 (Topic)」所組成。
- 每個主題都可以使用數個重要的「用詞 (Word)」來描述，且相同的用詞可同時出現在不同的主題之間。

應用 LDA 建立 topic model，LDA 是透過生成模型(觀察大量資料，估計出資料的生成機制)，在一系列文件中萃取出抽象的「主題」，LDA 假設了歌詞都是由少數幾個「主題 (Topic)」所組成，而且每個主題都可以由少數幾個重要的「用詞 (Word)」描述。

LDA 優缺點：快速、直觀且容易理解，且可用來預測沒看過的文件中的主題，而他的缺點是需要對模型做不少人為微調，模型的評估與驗證，及如何較有效的呈現結果以及提供更好的互動方式給消費使用者，都是未來研究發展的重點。

以下程式碼實現：LDA 模型，num_topics 設定主題的個數，透過 LDA 找到 5 個 topics。

圖 13　LDA 結果

五、模型架構

本計畫參考使用 Github 上 huwenxing 所提供的模型 TextCNN。

網址：https://github.com/649453932/Chinese-Text-Classification-Pytorch/blob/master/models

模型是用 TextCNN，其中參數有改，權重用 20 迭帶，學習率 0.003。

在模型的設計上使用的是 PyTorch，PyTorch 是一個開源的 Python 機器學習庫，基於 Torch，底層由 C++實現，應用於人工智慧領域，如自然語言處理。

主要有兩大特徵：

- 類似於 NumPy 的張量計算，可使用 GPU 加速。
- 基於帶自動微分系統的深度神經網路。

以下介紹模型的架構：

seq_len 是在數據處理中，將所有句子 padding 成一個長度。

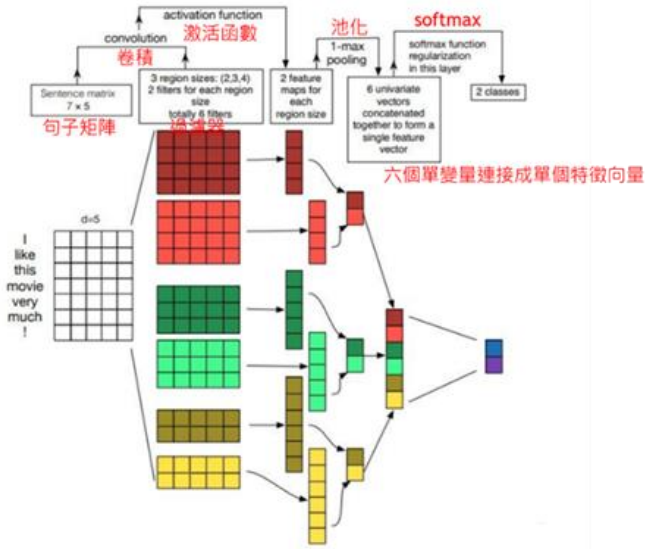


圖 14　TextCNN 模型架構圖

(一) 模型輸入：[batch_size, seq_len]

(二) 經過 embedding 層：加載預訓練詞向量或者隨機初始化,詞向量維度為 embed_size：[batch_size, seq_len, embed_size]

(三) 3.卷積層：在自然語言處理中卷積核寬度與 embed-size 相同，相當於一維卷積。3 個尺寸的卷積核：(2, 3, 4)，每個尺寸的卷積核有 100 個。卷積後得到三個特徵圖：[batch_size, 100, seq_len-1][batch_size, 100, seq_len-2][batch_size, 100, seq_len-3]

(四) 池化層：對三個特徵圖做最大池化。
[batch_size, 100][batch_size, 100][batch_size, 100]

(五) 拼接：[batch_size, 300]

(六) 全連接：num_class 是預測的類別數。[batch_size, num_class]

(七) 預測：softmax 歸一化，將 num_class 個數中最大的數對應的類作為最終預測 [batch_size, 1]

(八) 分析：卷積操作相當於提取了句中的 2-gram，3-gram，4-gram 信息，多個卷積是為了提取多種特徵，最大池化將提取到最重要的信息保留。

以下是程式碼的解釋：

安裝會使用到的套件以及工具包，以 PyTorch 搭建神經網絡，搭配 PyTorch 官方所提供的文件，設計所需的神經網路架構。torch 就是 PyTorch 的 package，大致上從創建到 training 的流程如下：

■ 載入訓練數據

數據集要先分好訓練集、測試集及驗證集,依照自身訓練比例的需求去分割。

■ 數據預處理

(一)圖片：統一大小、轉換色階等等

(二)文字：分詞、建立 word to id mapping、添加 token 等等

(三)打亂 data、batch data、轉乘 tensor 形式

■ 定義訓練過程

(一) 循環迭代（訓練過一整組資料集是一個迭帶，通常會訓練很多個迭帶）

(二) 批量加載數據

(三) 輸出，例如 loss、accuracy

(四) 儲存模型

■ 運行訓練腳本

Numpy 介紹

(一) 提供非常高效能的多維陣列(multi-dimensional array)數學函式庫

(二) 方便有用的線性代數(Linear Algebra)及傅立葉轉換(Fourier Transform)能力

(三) 利用 NumPy Array 替代 Python List

(四) 可定義任意的數據型態(Data Type)，使得能輕易及無縫的與多種資料庫

整合配置所需的參數，把訓練集、驗證集、測試集、分類的類別、詞性表、訓練結果、訓練詞向量等路徑設置好。

表 1　CNN 參數架構

參數名稱	參數值
dropout	0.5
require_improvement	1000
num_classes	len(self.class_list)
vocab	0
num_epochs	20
batch_size	128
pad_size	32
learning_rate	1e-3
filter_sizes	(2, 3, 4)
num_filters	256

六、訓練結果

(一) 雲端上操作

使用 Colaboratory，簡稱 Colab，是一個在雲端運行的編輯執行環境，由 Google 提供開發者虛擬機，並支援 Python 程式及機器學習 TensorFlow 演算法。若超過設的 1000 batch 效果未提升提早結束訓練，在迭帶 5800 的時候，1000 batch 效果未提升，提早結束訓練，全部測試損失值 0.3，準確率 90.95%，用時 58 分鐘。

表 2　雲端上結果

	precision	recall	f1-score	support
Sports	0.9248	0.8850	0.9044	1000
Politics	0.9286	0.9370	0.9328	1000
International	0.8627	0.8420	0.8522	1000
Finance	0.9483	0.9540	0.9511	1000
Fashion	0.8634	0.8660	0.8647	1000
Live	0.8915	0.9200	0.9055	1000
Society	0.8669	0.9050	0.8855	1000
3c	0.9558	0.9510	0.9534	1000
Games	0.9386	0.9020	0.9199	1000
Movies	0.9174	0.9330	0.9251	1000
accuracy			0.9095	10000
macro avg	0.9098	0.9095	0.9095	10000
weighted avg	0.9098	0.9095	0.9095	10000

(二) 本機端上操作

在迭帶器 5700 的時候，1000 batch 效果未提升，提早結束訓練，全部測試損失值 0.3，準確率 91.10%，用時 26 分鐘。

表 3　本機端上結果

	precision	recall	f1-score	support
Sports	0.9121	0.9030	0.9075	1000
Politics	0.9095	0.9450	0.9269	1000
International	0.8860	0.8470	0.8661	1000
Finance	0.9633	0.9440	0.9535	1000
Fashion	0.8443	0.8840	0.8637	1000
Live	0.8996	0.9140	0.9067	1000
Society	0.8864	0.9050	0.8956	1000
3c	0.9750	0.9360	0.9551	1000
Games	0.9295	0.9100	0.9197	1000
Movies	0.9111	0.9220	0.9165	1000
accuracy			0.9110	10000
macro avg	0.9117	0.9110	0.9111	10000
weighted avg	0.9117	0.9110	0.9111	10000

伍、結論

大數據資料必須要正確客觀，挑選出泛政治化的文章或者是資訊，達到反應事實，並且不要讓政治滲透進我們的討論，汙染到正確的思維，誤導大眾的觀念，甚至是有些人看文章標題就會被帶風向等行為。而本專題的目的就是利用臺灣各大社群網站文章(PTT 批踢踢實業坊、Dcard 等)作為資料來源，透過主題模型（Topic Model），結合深度學習與資料探勘(Text Mining)的方式應用在社群網路泛政治化留言之識別，藉由通過 CNN 卷積神經網路計算出來的詞向量，來判斷文本是否有泛政治化傾向，本研究判斷出來的結果將有利於了解於民眾或是當前社會的看法。

用上述的訓練方式重複訓練，有微調模型架構參數的卷積核的尺寸、學習率，進行每次訓練的紀錄(取比較好的)，之後進行比較，取比較好的模型運用在此專題上。而在研究中發現利用 TextCNN 模型架構在雲端以及本機端，在迭帶大約 5700 次的時候，batch 效果未提升，皆會提早結束訓練，測試損失值 0.3，都可以高達 90%以上的準確率，唯一比較大的差異是雲端的環境 GPU：NVIDIA TESLA V100，而在本機端環境是 CPU：Core i5-8250U，在訓練的時長差距一倍。未來改進會以設計新的 CNN 為目標，這樣所能控制的參數也會增多。

陸、引用文獻

[1]朱師右 (2020.08)。後疫情時代資訊服務產業的發展機會。工商時報。A9 版。

[2]全球 AI 人才趨勢：一探人工智慧、機器學習及深度學習之間的差異,(2020.01.14) , Retrieved from https://www.ecloudture.com/

[3]曾元顯 (2012.10)。圖書館學與資訊科學大辭典, Retrieved from http://terms.naer.edu.tw/detail/1679006/

[4]林婷嫻。斷開中文的鎖鍊！自然語言處理 (NLP), Retrieved from https://research.sinica.edu.tw/nlp-natural-language-processing-chinese-knowledge-information/

[5] Wikipedia contributors,(2015.10.21)自然語言處理, Retrieved from https://zh.wikipedia.org/wiki/[自然語言處理]

[6] itread01, (2018.12.22)概率主題模型簡介，LDA 基本思想, Retrieved from https://www.itread01.com/content/1545463463.html

[7] Wikipedia contributors,(2021.02.13)深度學習, Retrieved from https://zh.wikipedia.org/wiki/%E6%B7%B1%E5%BA%A6%E5%AD%A6%E4%B9%A0

[8] CKIP 中央研究院中文斷詞系統（2021/01/28）, Retrieved from http://ckipsvr.iis.sinica.edu.tw/

[9] Wikipedia contributors,(2020.05.06)泛政治化, Retrieved from https://zh.wikipedia.org/wiki/%E6%B3%9B%E6%94%BF%E6%B2%BB%E5%8C%96。

[10] Blei, David M., Ng, Andrew Y., Jordan, Michael I., & Lafferty, John. (2003). Latent Dirichlet Allocation, 3.

[11] Wikipedia contributors, (2020.12.10)TF-iDF, Retrieved from http://zh.wikipedia.org/zh-tw/Tf-idf

[12] Hu, Wenxing,(2020.04),Chinese-Text-Classification-Pytorch, Retrieved from https://github.com/649453932/Chinese-Text-Classification-Pytorch

[13] Pytorch, Retrieved from https://pytorch.org