「數據分析」
2016 年 12 月，11 卷 6 期，頁 35-62

基於 Python 之文字探勘平臺

林柏宇[1] 謝邦昌[2] 廖佩珊[3]

摘 要

隨著資訊科技的發展及手持裝置與社群網站越來越趨於活絡，各種電子新聞、社群網站的貼文與評論的資料量快速成長且結構複雜。一般而言，資料可簡單的分成結構化資料與非結構化資料，結構化的資料已有許多有效的方法可以運用，像是資料採礦技術，但如文字、聲音、影像等非結構化資料的分析方法，相較之下較為少數，運用本研究的文字探勘平台，挖掘出有效的資訊，將可以快速的從資料中探討其重要意義。

本研究希望透過網路上的開源碼整合出一套平台，利用 Python 做為後台運算，結合 HTML 撰寫網頁程式，把文字探勘的平台架在 Django 上。再將夏季旅展的新聞資料匯入平台，做文字探勘相關的分析，如詞雲分析、關聯分析、集群分析、情感分析等，討論夏季旅展資料的意義與脈絡。

關鍵字：文字探勘、大數據、情感分析、資料採礦

[1] 輔仁大學 統計資訊學系應用統計碩士班
[2] 台北醫學大學 大數據研究中心及管理學院
[3] 輔仁大學 統計資訊學系

36
Journal of Data Analysis
Dec 2016, Vol.11 No.6, pp.35-62

Text mining platform with python

Po-Yu Lin [1] Ben-Chang Shia [2] Pei-San Liao [3]

Abstract

With the development of information technology, handheld devices and social networking sites become more and more active, a variety of electronic news and community website postings and comments rapidly growing amount of data and complex structure. In general, the data can be simply divided into structured data and unstructured data, structured data there are many effective methods can be applied, such as data mining technology. But such as text, sound, video and other unstructured data analysis method, compared to relatively few, in this study the use of text mining platform, found out an effective information, will be able to quickly explore its significance from the data.

We hope that through this study, an open source web platform for the integration of a set, use Python as a background operation, combined with HTML pages written program, the text mining platform on the shelf in Django. Then TTE news data import platform, do text mining-related analysis, such as word cloud analysis, correlation analysis, cluster analysis, sentiment analysis, etc., to discuss the meaning and context of information TTE.

Keywords: Text Mining, Big Data, Semantic analysis, Data Mining

壹、緒論

一、研究背景與動機

隨著資訊科技的發展及手持裝置與社群網站越來越趨於活絡，各種電子新聞、社群網站的貼文與評論的資料量快速成長且結構複雜，使得傳統的分析方法受到限制。一般而言，資料可簡單的分成結構化資料與非結構化資料，結構化的資料已有許多有效的方法可以運用，像是資料採礦技術，但如文字、聲音、影像等非結構化資料的分析則需要用到文字探勘的技術，以挖掘出有效的資訊，提供企業做決策的參考。

近年來大數據的話題持續升溫，文字探勘技術的應用就因應而生，尤其是 Facebook、Twitter、部落格文章的興起，使用者能隨時隨地發表自己的意見，而這些都是非結構化的文字資料，該如何在眾多的篇章中，快速的找出文章中想表達的重點，而不需要一篇一篇的去閱讀，這時候就需要使用到文字探勘的技術了。

二、研究目的

文字探勘技術在網路上有許多開源碼供大家使用，但大多都是片段式的，本研究希望透過網路上的開源碼整合出一套平台，利用 Python 做為後台運算，結合 HTML 撰寫網頁程式，把文字探勘的平台架在 Django 上，讓使用者可以直接連網運用。使用者將網頁爬下的文本，透過平台做文字探勘的分析，最後運用資料視覺化呈現出來，讓使用者有一整套的平台服務，作為決策的參考。

三、研究流程

本研究的研究流程包含五個部分，第壹章緒論，包含研究背景與動機、研究目的、研究流程等三節；第貳章為文獻探討，包含文字探勘、資料採礦、資料視覺化等相關文獻進行探討；第參章為研究方法，包含研究架構、文字探勘技術、研究工具等進行，第肆章為實證分析；第伍章為結論與建議。研究流程圖如圖 1-3-1 所示。

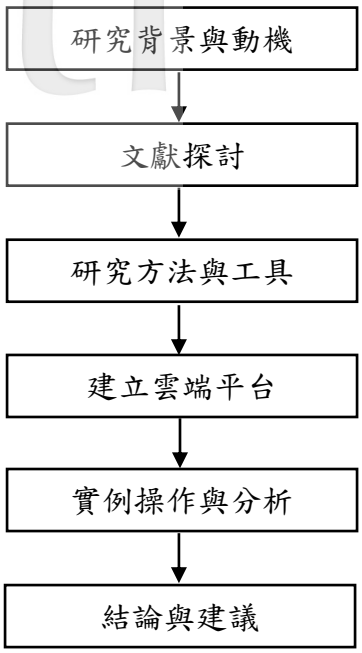

圖 1-3-1 研究流程圖

貳、文獻探討

一、文字探勘

(一)文字探勘定義

資料探勘(Data Mining)與文字探勘(Text Mining)的關係緊密，前者是處理結構化的數值型資料型態，而後者是針對文字進行分析，在處理非結構化與半結構化的資料型態中，挖掘出隱含在文字中有用的訊息。

Sullivan(2001)定義文字探勘為一種編輯、組織及分析大量文件的過程，用以符合使用者的特定資訊需求及發現某些特徵之間的關聯。巫啟台(2002)將文字探勘定義為「從非結構化的文字中發掘出有用或是有趣的片段、模型、方向、趨勢或規則」。謎家蘭(2006）所有文件的分佈提供一個總覽，以提升文件的搜尋效益，並自動建立文件的分類架構，辨識文件中的字詞與關聯性，以減少文件檢索和查詢的誤判。

Fayyad(1996)在非結構化或半結構化的文字資料中，使用資訊檢索(Information Retrieval, IR)、資訊萃取(Information Extraction, IE)與自然語言處理(Natural Language Processing, NLP)，進一步挖掘出尚未得知的訊息。Hearst(1999)提出文字探勘的定義是從文獻中擷取隱含的知識，以簡要的格式呈現資料給使用者。

(二)文字探勘技術之相關文獻

林名彥(2015)網際網路的盛行下許多消費者會透過網路論壇來發表意見，尤其是網購商品的抱怨；目前企業對於顧客抱怨(又稱客訴)的處理，大多是以客戶服務中心人員來取得顧客抱怨資訊而進行處理，對於網路論壇上的抱怨資訊常常是無法來處理。因此，本研究搜集網路論壇的客訴文章進行文字探勘，以尋找抱怨文中的關鍵字詞，並瞭解網友們經常抱怨的主題和關聯的字詞。

鄭凱文(2014)本研究樣本為 2011 年中國大陸所有上市公司所揭露的 MD&A 及相關財務資訊，MD&A 非量化資訊係運用 Stanford Word Segmenter 斷詞資料庫、正負向詞典、TFIDF、K-means 等技術進行群集分析，並結合財務資訊的 K-Means 群集分析，分析出中國大陸 2011 年上市公司 MD&A 揭露是否誇大。

劉育華(2014)本研究以兩家宮廟的 Facebook 粉絲專頁的官方貼文為分析標的，以文字探勘(Text Mining)的工具從貼文中找出最常出現的關鍵字，分析其詞頻以文字雲(Word Cloud)來呈現，並依照內容分析法(Content Analysis)將關鍵字貼文做性質分類，進一步使用社會網絡分析(Social Network Analysis)，來探討哪些類型及性質的貼文會吸引較多使用者回應，期望藉由文字探勘及社會網絡的結合，找出使用者最感興趣的主題與溝通方式

陳譽晏(2015)在大量的資料中，存在著數字型態的結構化資料和文字、聲音、影像的非結構化資料。本研究先利用 Ubuntu 進行平臺的架設，再利用 R Shiny 建構文字探勘平臺。接著將台積電的新聞資料，予以匯入文字探勘平臺中，並跑出一連串之分析，如詞雲分析、集群分析、脈絡分析、關聯分析、情感分析和動態圖表等，從文字探勘的分析方法中，探討匯入的文本資料之意義。

陳柏江(2014)本研究提出以文字探勘結合推薦系統的方式，蒐集關於失智症之文字資訊，進行分類並依照使用者正在閱讀的文章以及感興趣的領域進行文章的推薦，為非專業的照護人員建立一個知識分享的平臺。一方面提升使用者對於失智症照護活動的知識，另一方面也減少使用者在資訊蒐集上的白做工。此外，在推薦系統流程上將導入文字探勘所用之 TF-IDF 方法，產生「同中求異」的推薦成果，讓使用者能接受到雖為同一議題但有不同方針或不同看法的文章。

吳宜隆(2010)本研究主要目的在建置一個以雲端運算為基礎之非結構化文字資料之探勘服務；讓使用者得以透過網頁來進行登入、執行、監控與瀏覽的服務平臺。

Francis(2006)文字探勘者需要學習在各個領域的研究中有很多運用了文字探勘技術去挖掘出有效的訊息，但是將文字探勘技術架設在平臺上的研究並不多見，因此本研究希望能建立文字探勘平臺，提供給各領域的人使用。

二、資料採礦

(一)資料採礦的定義

資料採礦是指在資料庫中，利用各種分析方法與技術，將過去累積的大量繁雜的歷史資料中，進行分析、歸納與整合等工作，以萃取出有用的訊息，找出有意義且使用者有興趣的特徵，提供企業做為決策的參考依據。

資料採礦為尋找資料中隱藏的訊息，如趨勢（Trend）、特徵（Pattern）及相關性（Relationship）的過程，也就是從資料中挖掘出資訊或知識（Knowledge Discovery in Databases, KDD），也有人稱為資料考古學(Data Archaeology)、資料特徵分析(Data Pattern Analysis)或功能相依分析(Functional Dependency)目前已被許多研究人員使用，視為結合資料庫系統和機器學習技術的重要領域，許多產、業界人士也認為資料採礦為企業使用指標中，重要的一環（謝邦昌、蘇志雄、鄭宇庭，2011）。

(二)資料採礦的功能

資料採礦的功能，主要包含五項功能，分類（Classification）、推估（Estimation）、預測（Prediction）、關聯分組（Affinity grouping），和同質分組（Clustering），詳細內容如下所示：

1. 分類（Classification）

按照分析對象的屬性分門別類加以定義，建立類組(Class)。例如，將股票購買的風險屬性，區分為高度風險申請者，中度風險申請者及低度風險申請者。使用的技巧有決策樹（Decision tree）、羅吉斯迴歸（Logistic regression）、記憶基礎推理(memory-based reasoning)等。

2. 推估 （Estimation）

推估為預測連續值之相關屬性資料。例如，按照購買股票者之教育程度、行為別來推估其購買股票之消費量。使用的技巧包括統計方法上之相關分析、迴歸分析（Linear Regression）及類神經網路（Neural network）等方法。

3. 預測（Prediction）

從所有物件決定那些相關物件應該放在一起。例如超市中相關之盥洗用品 (牙刷、牙膏、牙線)，放在同一間貨架上。在客戶行銷系統上，此種功能係用來確認交叉銷售 (cross-selling) 的機會以設計出吸引人的產品群組。

4. 關聯分組（Affinity grouping）

由資料裡的所有變數中，決定那些相關變數應該放在一起。例如，賣場中相關聯的物品會擺放在一起。在消費者顧客分析上，被用來交叉銷售上，以設計出會讓消費

者購買的產品組合。

5. 同質分組（Clustering）

在不同的母體中區隔為較具同類型之群體(Clusters)。同質分組相當於行銷術語中的區隔化(Segmentation)。目的是能將未處理的資料能依同類型做分組，使分析者能更瞭解資料的特性，使用方法包括 K-means 法及 Agglomeration 法。

(三)資料採礦交叉行業標準過程(CRISP-DM)

CRISP-DM 起源於 1990 年，由 SPSS 和 NCR 兩大廠商在合作 Daimler Benz 的案子中，進行資料倉儲和資料探勘過程中發展出來，而 CRISP-DM 的簡稱為「Cross-Industry Standard Process for Data Mining」。目前 CRISP-DM 模型為該小組在 1997 年到 1999 年研究之後，於 2000 年提出的資料採礦標準化作業流程。在整體規劃設計後，於 2000 年推出 CRISP-DM 1.0 模型，藉由實證分析，把資料採礦過程中所需的步驟都加以標準作業化，需要對企業的需求及主要問題進行瞭解，以及後期對模式的評價與模式的延伸應用都是一個完整的資料採礦過程不可或缺的要素。CRISP-DM 是從方法學的角度強調實施資料採礦專案的方法和步驟，並獨立於每種具體資料採礦演算法和資料採礦系統（謝邦昌，2011）。

1. 瞭解企業需求 (Business Understanding)

主要是以企業的觀點來找出執行此方案的主要目的，在此步驟要先定義資料採礦的問題，並且訂定初步研究計畫。

2. 瞭解資料特性 (Data Understanding)

收集到完整資料後，並對收集的資料作初步分析，包括資料預處理、識別資料的質量問題、找到對資料的基本觀察，接著並設立假設前提。

3. 準備資料 (Data Preparation)

篩選資料中各項表格、紀錄以及主要變數，接著處理經篩選出來的資料，包括變數、數字資料等，即可應用於模型選擇工具上。

4. 設計模型 (Modeling)

此步驟著重於選擇並應用一或多種資料探勘技術，如關聯分析演算法（Association）、群集演算法（Cluster）、決策樹演算法（Decision tree）、線性迴歸演算法（Linear Regression）、羅吉斯迴歸演算法（Logistic regression）、貝氏機率分類演算法（Naive Bayes）、類神經網路演算法（Neural network）、時序群集演算法（Sequence Clustering），和時間序列演算法（Time Series）等。

5. 評估 (Evaluation)

主要是分析結果，並證實前一步驟設計的模型是否符合企業所需，而達到推動方案之目的，以及進一步的決定將來是否繼續採用此一模型。

6. 建置 (Deployment)

此步驟主要是經各種評估後，如模型的可適性。若所建立之模型符合企業目標，則將再進一步擬定該模式之推動計畫。

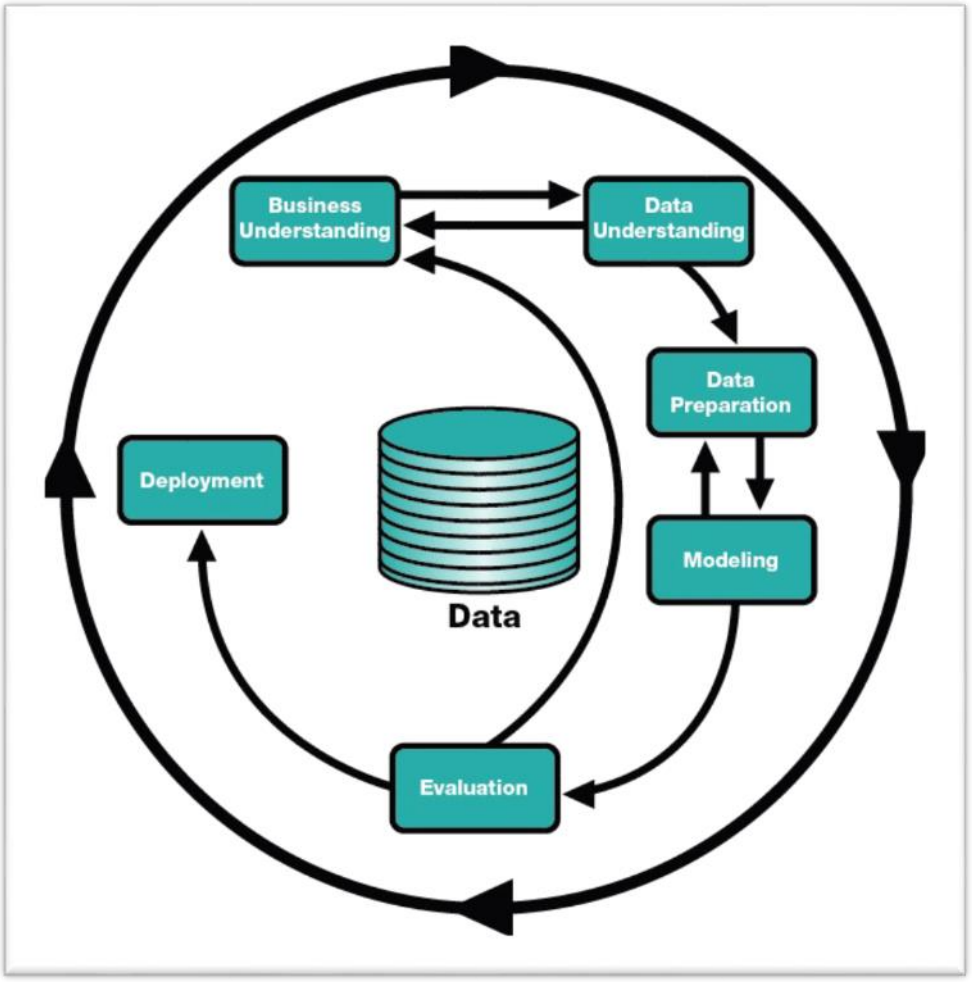

圖 2-2-1 CRISP-DM PROCESS

資料來源：http://www.sv-europe.com/crisp-dm-methodology/

三、資料視覺化

在 Big Data 越來越重要的時代下，資料量越來越多，資料的繁雜程度也越來越高，該如何清楚又簡潔的呈現這些資料，就要運用到資料視覺化的技術，讓數字會說話也能看圖說故事，使我們挖掘出來的資料完整又漂亮的呈現。

陳芸芸(2004)許多研究顯示，越來越多的年輕人對文字的依賴與閱讀習慣，逐漸被圖像所取代。張文瑜(2005)將量化的資料結果，以圖像呈現資料中的某些特性，取代資料中的數字表示，就是資料視覺化。Messaris (1994)容易先注意到傳播圖片的影像，而比較會忽略閱讀傳播文字或廣告文案。楊尊宇(2015) 資料視覺化能幫助我們以直觀的方式處理海量資料。

資料視覺化的三個關鍵要素：

1.資料的正確性-即便在極簡的思維中，資料都是最重要的。務必要提供正確有效的資料

2.理解資料的動機-使讀者跨過心理門檻，願意深入理解資料的意涵

3.資訊傳遞的效率-降低讀者吸收資訊的難度，更有效的讓資訊被傳播出去

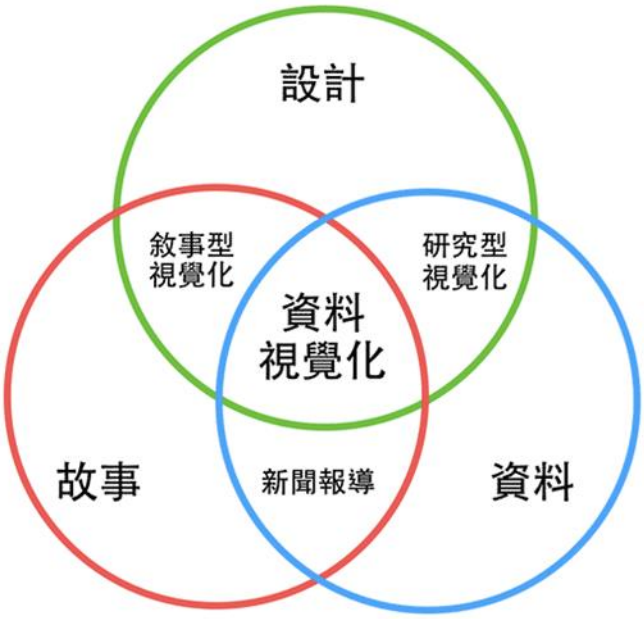

圖 2-3-1 資料視覺化結構

資料來源：http://blog.infographics.tw/2015/06/three-keys-to-visualization/

由於人類無法在繁雜的數字中第一時間判斷出資料的趨勢，所以使用圖像的表達會更容易去掌握資料的形態，而資料視覺化運用顏色、色塊、線條、動態等來呈現，可以讓資料的訊息傳達的更順暢，所以資料視覺化在資料中的呈現會越趨重要。

參、研究方法

一、研究架構

本研究利用 Django 與 Python 建構雲端平臺；之後利用此文字探勘平臺讀取資料，進行一連串的資料分析，如：資料預處理、詞雲分析、集群分析、脈絡分析等；再將輸出之結果以視覺化圖型呈現；最後得出結論。研究架構如圖 3-1-1。

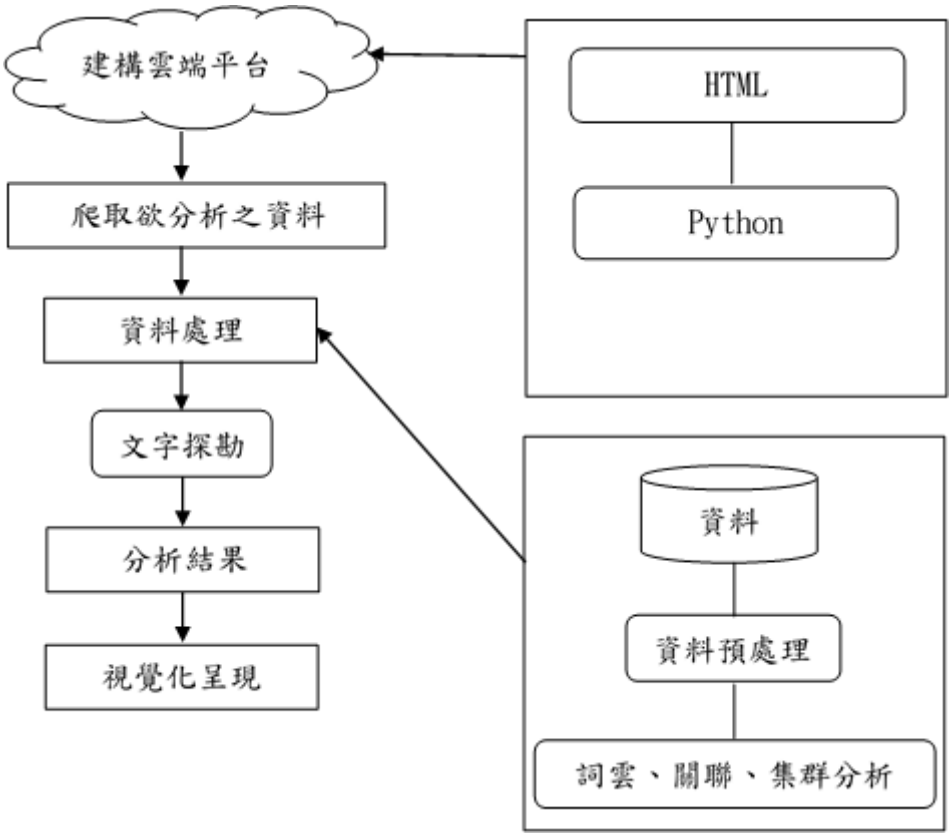

圖 3-1-1 研究架構圖

二、文字探勘技術

由於資料的型態為非結構化的數值型態，而是長短不一的非結構化文字資料，所以在進行文字探勘之前必須先將資料做預處理，將資料整理成適合後續處理的格式，再利用詞頻計算與權重來分析詞語之間所產生的價值。

(一) 文本預處理

從網路上所爬下來的資料可能包含了很多贅詞、亂碼與廣告，或是雜亂無章的篇幅，因此必須先對所要處理的檔進行分割整理，將研究的目標進行分類，這樣可以有效的進行後續的分析。

(二) 中文斷詞

詞是最小有意義且可以自由使用的語言單位。任何語言處理的系統都必須先能分辨文本中的詞才能進行進一步的處理，例如機器翻譯、語言分析、資訊萃取。由於中文詞集是一個開放集合，不存在任何一個詞典或方法可以盡列所有的中文詞。當處理不同領域的文檔時，領域相關的特殊詞彙或專有名詞，常常造成分詞系統因為參考詞彙的不足而產生錯誤的切分。為瞭解決這個問題，最有效的方法是補充領域詞典加強詞彙的搜集。

中文與英文最大的差異在於中文的詞可以由一個或是兩個以上鄰近的字組成，而詞與詞之間不像英文有明顯的區隔。此時我們就需要透過中文斷詞的過程來將文章中的句子分成詞。在台灣方面，是由中央研究院資訊所的詞庫小組，所建立的線上詞庫系統，可以做中文的斷詞斷句，也可以新增詞庫。中文斷詞分成三種方法，詞庫式斷詞法、N 元斷詞法以及混合式斷詞法。

詞庫式斷詞法主要是針對某個領域來建構出所相對應的詞庫，利用詞庫的內容與字串的比對，萃取出相對應的字詞，但是並不會把所有的字詞都考慮進去；且詞庫在新增字詞上，為了要保持它的準確度，有時必須要用人工去判斷，所以在詞庫的維護上有很大的困度。

N 元斷詞法主要是利用各種字串組合，像是二元或是三元，若是要取出更長的字詞就要新增新的組合，像是五元或是六元等；而 N 元斷詞法可以不用依賴詞庫就可以找出許多字詞，因此時常被用在詞庫的新詞萃取上；但是有時萃取出來的字詞並沒有意義，會使用統計方法來判定是否為字詞，將其優化。

混合式斷詞法則是結合前述兩種斷詞法的方法。先利用詞庫式斷詞找出許多不同組合的字詞，再利用 N 元斷詞法判定字詞的字串找出最佳的斷詞組合，此法目前仍需要大型的語料庫來提供統計資訊。

(三) 詞頻與權重計算

一個字詞在某個文件，或是語料庫中出現的次數，稱之為詞頻(Term Frequency，TF)。在詞頻的基礎上，要對每個字詞分配一個權重，例如在中文中最常見的詞，如「的」、「是」、「在」、「要」等，這些詞給予較小的權重，而較少出現的詞，如「電腦」、「手機」等，則給予較大的權重。這樣的權種稱為逆像文件頻率(Inverse Document Frequency，IDF)，而逆向文件頻率的大小與一個字詞的常見程度成反比。

TF-IDF(Term Frequency-Inverse Document Frequency)是一種用於文字探勘常用的技術，它是一種統計方法，用以評估一個字詞對於一個文件的重要程度。主要的概念為某一個字詞在文件中出現的詞頻(TF)高，而在其他文件中出現的少，則這個詞具有不錯的區別能力。

TF-IDF 法的公式定義如式：

TF-IDF(i)=TF(W_i,d_j)×IDF(W_i)

=TF(W_i,d_i)×log(D/(DF(W_i)))

其中 TF(W_i,d_j)表示字詞 W_i 在文件 d_i 中所出現的頻率，D 代表總檔數，DW(W_i)表示包含字詞 W_i 之文件數。

(四) 關聯分析

由上述的詞頻計算後，我們可知道那些字詞是出現頻率最高的，將這些字詞做關聯分析，再找出相關性高的字詞或是我們想研究的目標字詞的相關性。

相關係數(Correlation coefficient)，用以反映兩個變數之間的相互關係及相關方向，公式如：

$$r = \frac{\sum_{i=1}^{n}(x_i-\bar{X})(y_i-\bar{Y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{X})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{Y})^2}}，其中|r| \le 1 \ ，i = 1,2,3 \dots n \qquad (3.1)$$

1. 相關係數的值是沒有單位的。
2. 當 r>0 時，表示兩變數正相關，r<0 時，兩變數為負相關。
3. 當|r|=1 時，表示兩變數為完全線性相關，即為函數關係。
4. 當 r=0 時，表示兩變數間無線性相關關係。
5. 一般可按三級劃分：|r|<0.4 為低度線性相關；0.4≤|r|<0.7 為顯著性相關；0.7≤|r|<1 為高度線性相關。

(五) 集群分析

集群分析是一種利用邏輯程式，來探討如何將欲測對象分為相類似的群體。主要的目的在於辨識相似特性的事物，再按照這些特性去分群，同一個集群內的的事物有較高的同質性，而不同集群的事物有較高的相異性。集群分析按照分類方法的不同可分為階層群集分析法與非階層集群分析法。

1. 階層式集群法

階層式集群法透過一種階層架構的方式，將資料層層反復的進行分裂或聚合，以產生最後的樹狀結構，常見的方式有兩種；如果採用聚合的方式，階層式分群法可由樹狀結構的底部開始，將資料或群聚逐次合併；如果採用分裂的方式，則由樹狀結構的頂端開始，將群聚逐次分裂。

階層式集群法基本有四種集群間距離的計算方式，分別為單一連結聚合演算法、完全連結聚合演算法、平均連結聚合演算法、華德法，以下對四種方法做個別介紹：

(1) 單一連結聚合演算法(Single Linkage agglomerative algorithm)

集群與集群間的距離可以定義為不同集群中最接近兩點間的距離，公式如。

$$\mathrm{d}(c_i, c_j) = \min_{a \in c_i, b \in c_j} d(a, b)$$

其中 d(a,b)表示 c_i 群內第 a 樣本與 c_j 群內第 b 樣本之距離。

(2) 完全連結聚合演算法(Complete Linkage agglomerative algorithm)

集群間的距離定義為不同集群中最遠兩點間的距離，公式如。

$$\mathrm{d}(c_i, c_j) = \max_{a \in c_i, b \in c_j} d(a, b)$$

其中 d(a,b)表示 c_i 群內第 a 樣本與 c_j 群內第 b 樣本之距離。

(3) 平均連結聚合演算法(Average Linkage)

集群間的距離定義為不同集群間各點與各點間距離總合的平均，公式如。

$$\mathrm{d}(c_i, c_j) = \frac{\sum_{i \in c_i} \sum_{j \in c_j} D_{ij}}{n}$$

其中 D_ij 表示 c_i 群內第 i 樣本與 c_j 群內第 j 樣本之距離，n 為全部距離的個數。

(4) 華德法(Ward’ s method)

集群間的距離定義為在將兩群合併後，各點到合併後的群中心的距離平方和，公式如。

$$\mathrm{d}(c_i, c_j) = n_{c_i} \times \left|\bar{x}_{c_i} - \bar{\bar{x}}\right|^2 + n_{c_j} \times \left|\bar{x}_{c_j} - \bar{\bar{x}}\right|^2$$

其中n_{c_i}表示c_i群的樣本數、n_{c_j}表示c_j群的樣本數，$\bar{\bar{x}}$表示兩群合併中心點。

2. 非階層式集群法

在所有的非階層式集群演算法中，K-means 是最典型的方法。在使用此方法之前，必預先決定分群結果的集群數量，也就是定義 k 的值。當 K-means 初始化時，會先任意選擇 k 個資料點做為集群的中心點。

K-means 以集群的中心點來代表所有資料點，所以能減少大量的計算，但是隨機選擇的初始中心點不恰當時，會造成分群效率不佳，降低分群可靠度。而且該方法以群聚的重心作為集群的代表點，所以集群結果很容易被雜訊(Noises)或是離群值(Outliers)所影響，而且無法辨識出非凸邊形的集群。K-Means 集群法計算公式如。

$$J = \sum_{j=1}^{K} \sum_{i=1}^{n} \left| x_i^{(j)} - c_j \right|^2$$

其中$|x_i{}^{(j)} - c_j|^2$為K群裡第j群中的第i個樣本與該群中心點c_j之距離平方。

三、研究工具

(一) Python

1. Python 介紹

Python 的創始人為 Guido van Rossum。1989 年的聖誕節期間，Guido van Rossum 為了在阿姆斯特丹打發時間，決心開發一個新的腳本解釋程式，作為 ABC 語言的一種繼承。Python 的官方直譯器是 C-Python，該直譯器用 C 語言編寫，是一個由社群驅動的自由軟體，目前由 Python 軟體基金會管理。

2. Python 特色

(1)基本概念

Python 是一種物件導向、直譯式的電腦程式語言，具有近二十年的發展歷史。它包含了一組功能完備的標準庫，能夠輕鬆完成很多常見的任務。它的語法簡單，與其它大多數程式設計語言使用大括弧不一樣，它使用縮進來定義語句塊。Python 具備垃圾回收功能，能夠自動管理記憶體使用。它經常被當作腳本語言用於處理系統管理任務和網路程式編寫，然而它也非常適合完成各種高階任務。

(2)物件導向

Python 是完全物件導向的語言。函式、模組、數字、字串都是物件。並且完全支援繼承、重載、衍生、多重繼承，有益於增強原始碼的重複使用性。Python 支援重載運算符，因此 Python 也支援泛型設計。

(3)可擴展性

Python 本身被設計為可擴充的。並非所有的特性和功能都整合到語言核心。Python 提供了豐富的 API 和工具，以便程式設計師能夠輕鬆地使用 C、C++、C-Python 來編寫擴充模組。Python 編譯器本身也可以被整合到其它需要腳本語言的程式內。因此，有很多人把 Python 作為一種「膠水語言」(glue language) 使用。使用 Python 將其他語言編寫的程式進行整合和封裝。在 Google 內部的很多專案，例如 Google App Engine 使用 C++編寫效能要求極高的部分，然後用 Python 或 Java 調用相應的模組。

(4)可攜性

Python 的標準實現是由可移植的 ANSI C 編寫的，可以在目前所有的主流平臺上編譯和運行。Python 程式的核心語言和標準庫可以在 Linux 、Windows 和其他帶有 Python 解釋器的平臺無差別的運行。大多數 Python 外圍介面都有平臺相關的擴展，但是核心語言和庫在任何平臺都一樣。Python 還包含了一個叫做 Tkinter 的 Tk GUI 工具包，它可以使 Python 程式實現功能完整的無需做任何修改即可在所有主流 GUI 平臺運行的用戶圖形介面。

(5)標準庫介紹

python 的強大功能之一是帶有一個非常完全的標準庫,通過該標準庫,我們可以方便地實現大量功能。Python 具有豐富和強大的類庫，其標準庫包括了很多的模組, 從 Python 語言自身特定的類型和聲明, 到一些只用於少數程式的不著名的模組。任何大型 Python 程式都有可能直接或間接地使用到這類別模組的。例如：

Sys 模組：此模組包含系統對應的功能。對於有經驗的程式師，sys 模組中其他令人感興趣的項目有 sys.stdin、sys.stdout 和 sys.stderr 它們分別對應你的程式的標準輸入、標準輸出和標準錯誤流。

OS 模組：這個模組包含普遍的作業系統功能。如果你希望你的程式能夠與平臺無關的話，這個模組是尤為重要的。即它允許一個程式在編寫後不需要任何改動，也不會發生任何問題，就可以在 Linux 和 Windows 下運行。

(二) Django

1. Django 介紹

Django 是一個用 Python 所寫的高階網頁框架，它可以讓人快速的開發實用又乾淨的網頁。Django 是兩年前因應某一個線上新聞網站的運作開發而成的，它的設計主要是為了能處理密集的新聞資料，以及讓網站開發者可以在最短的時間內看到網站開發的內容。Django 最重要就是可以讓你快速開發一個高效能及精緻的網站。

Django 有自己的一個 HTTP server，使用者從瀏覽器透過 HTTP server 來要求資料時，server 會對照 urls.py 來找到相對映的 Python 程式來處理。在處理的過程當

中，通常我們會讀某一個 template 檔，而此時 Django 就會將 template 所需要的資料都處理過後，再做回應。

2. Django 特色

Django 介紹

(1).MVC 設計模式

MVC (Model-View-Control) 是一種軟體發展的方法，它把代碼的定義和資料訪問的方法（模型）與請求邏輯 （控制器）還有使用者介面（視圖）分開來。

這種設計模式關鍵的優勢在於各種元件都是鬆散結合的。這樣，每個由 Django 驅動 的 Web 應用都有著明確的目的，並且可獨立更改而不影響到其它的部分。例如，開發者更改一個應用程式中的 URL 而不用影響到這個程式底層的程式。設計師可以改變 HTML 頁面的樣式而不用接觸 Python 代碼。資料庫管理員可以重新命名資料表並且只需更改專屬的位置，無需從一大堆文檔中進行查找和替換。

(2).Template 模板

有別於原始的 PHP 寫法，Django 使用模板來呈現網頁的內容。模板大部分都是 HTML 的程式碼，只有部份的資料會用特殊的 Django template 格式來取代，當然還有一些控制的機制，例如 for 迴圈、if else 判斷等。這樣一來，網頁設計的部份就和資料的處理分離，不論是資料處理或是網頁的部份，程式碼看起來都會乾淨許多。

(3).資料庫

Django 支援 4 種資料庫，包括 PostgreSQL、SQLite3、MySQL、Oracle，Django 內建的資料庫是 SQLite3，這 4 種資料庫都可以與 Django 框架有良好的互動，建議使用 PostgreSQL，它在成本、特性、速度和穩定性方面都比較平衡。

(三) 研究工具架構

由 URL 發出需求給 View，View 再向 Model 與資料庫撈取所需要的資料，撈取完資料後，將資料回傳給 View，最後藉由 Templates 模板在瀏覽器上呈現。

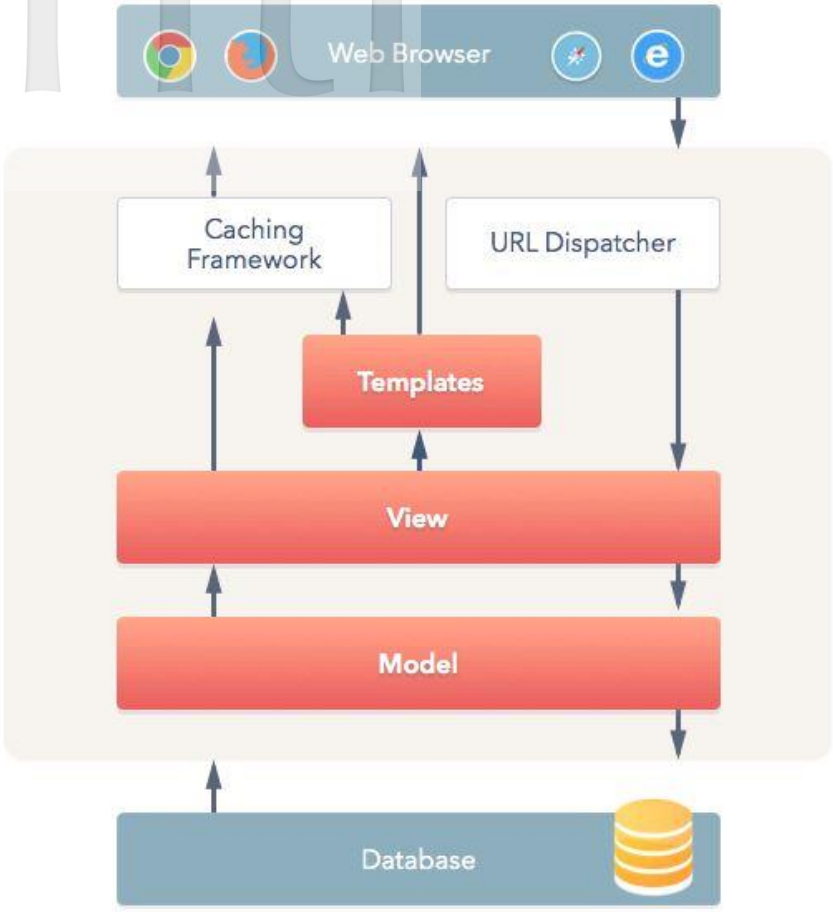

圖 3-3-1 研究工具架構圖

肆、實證分析

在本章實證分析中，由於文字探勘技術運用在旅展的研究並不多，所以本研究將以「台北夏季旅展」為主題來做探討。本研究分別在 ETtoday 東森新聞雲和中時新聞電子報中抓取和「台北夏季旅展」相關之新聞，透過 Python 進行新聞的爬蟲，其新聞篇數如表 4-1 所示。資料時間為 2014 年 5 月至 2014 年 6 月與 2015 年 5 月至 2015 年 6 月，將探討台北夏季旅展在這兩年之間的差異，並找出其相關之重要因子，最後給出適當的結論與建議。

表 4-1 夏季旅展新聞篇數

年份	新聞篇數
2014 年	20 篇
2015 年	20 篇

一、分析內容

(一) 詞雲分析

在 2014 年與 2015 年的詞雲中，可以明顯的看到「旅展」，在 2014 年裡，有出現台北、優惠、夏季、住宿、博覽會、國際觀光、旅遊、免費等；在 2015 年裡，有出現台北、優惠、團費、自由行、東森、旅遊、飯店、機票等。

圖 4-1-1 詞雲分析-2014 年

圖 4-1-2 詞雲分析-2015 年

(二) 關聯分析

找出與「旅展」相關聯的字詞，並探討旅展在 2014 年與 2015 年的這兩年中，所發生的事件。如表 4-1-1，在 2014 年，「優惠」的關聯性最高，其次是旅遊、台北；在 2015 年，也是「優惠」的關聯性最高，其次是台北、自由行。

表 4-1-1 與「夏季旅展」相關聯的字詞

年份	與「旅展」相關聯字詞				
2014 年	優惠(0.76)	旅遊(0.71)	台北(0.66)	旅行社(0.62)	夏季(0.62)
	推出(0.56)	博覽會(0.54)	機票(0.48)	飯店(0.41)	觀光(0.32)
2015 年	優惠(0.73)	台北(0.71)	自由行(0.62)	旅遊(0.59)	團費(0.58)
	博覽會(0.53)	住宿(0.53)	含稅(0.42)	推出(0.38)	現場(0.31)

(三) 集群分析

集群分析是以距離與相似的程度做為分類的依據，相似度越高的就會被歸類到同一類。在 2014 年的集群分析中可以看到，優惠、飯店為同一類，國際為一類，世貿、台北、攤位等為同一類，觀光、今天、推出為同一類。

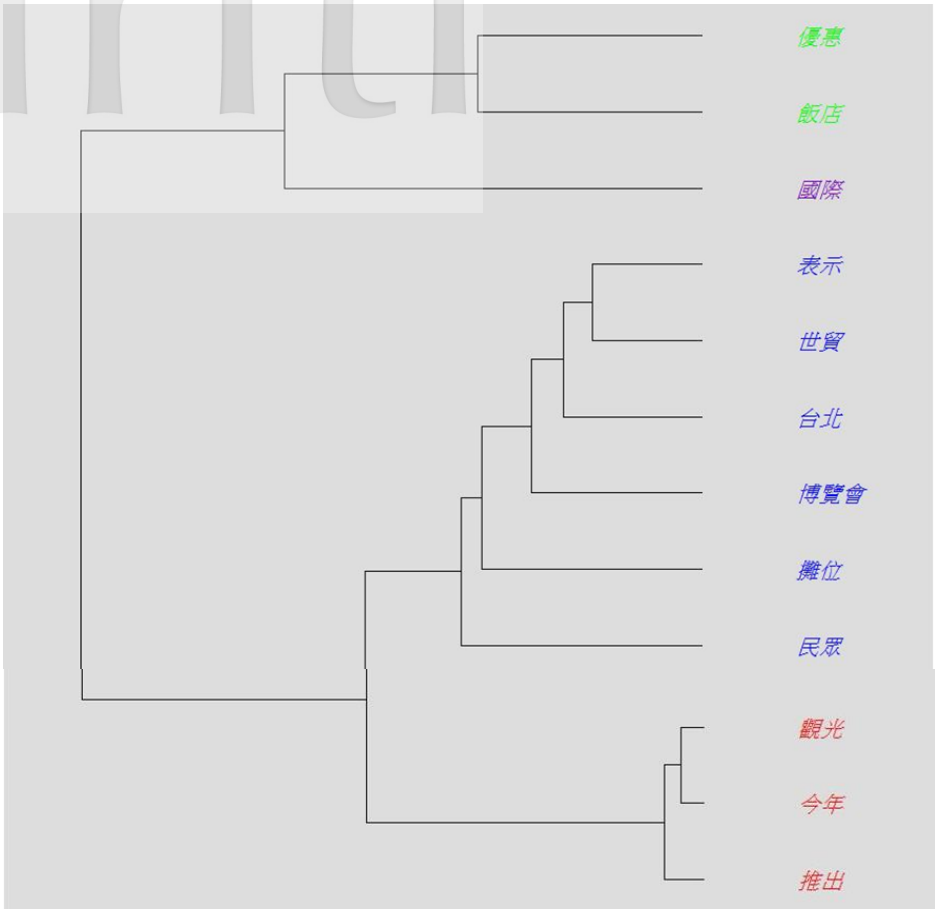

圖 4-1-3 集群分析-2014 年

在 2015 年的集群分析中，民眾、日本、現場為一類，萬元、自由為一類，優惠、國際、推出、台北、觀光為一類。

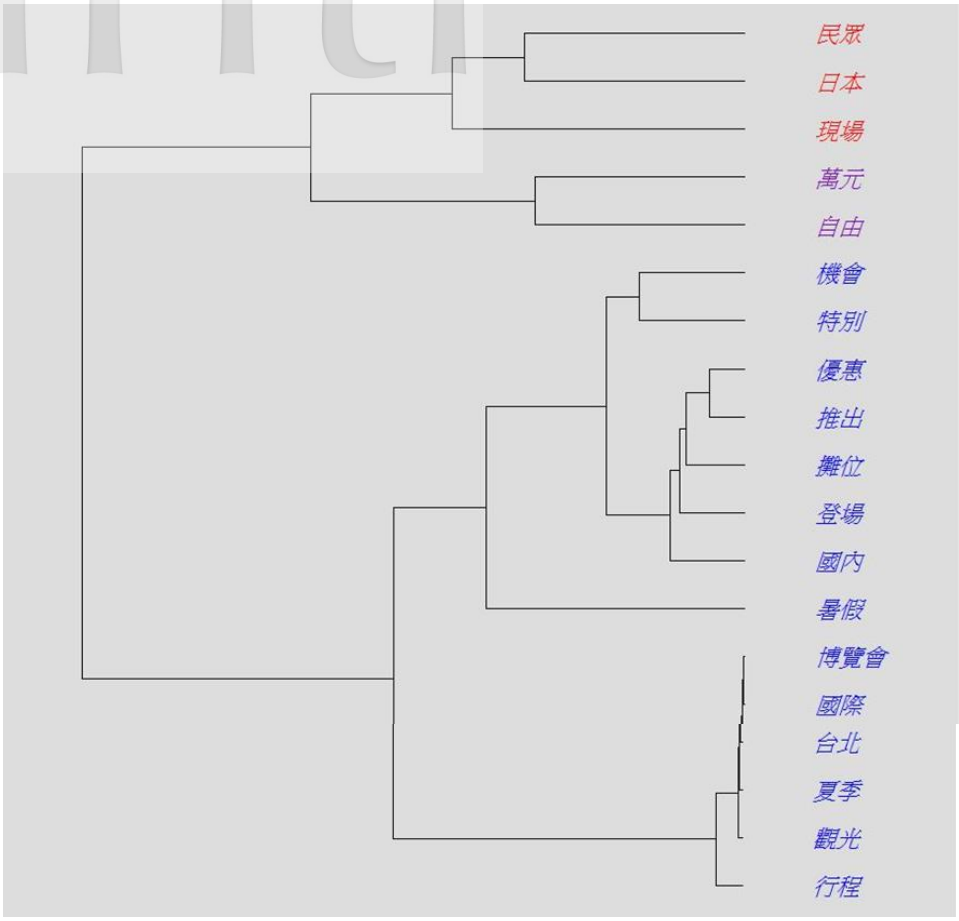

圖 4-1-4 集群分析-2015 年

(四) 脈絡分析

文本背後所蘊含的意義通常有其建構論述的脈絡，為了從 2014 與 2015 年夏季旅展中的新聞篇章中，快速的了解到這兩年的重要資訊，利用脈絡分析可以簡略得知每一條脈絡的始末。在 2014 年的脈絡分析為，旅展→旅遊→台北→夏季→業者→住宿→國際觀光；在 2015 年的脈絡分析為，旅展→旅遊→優惠→自由行→博覽會→機票→國際觀光。

圖 4-1-5 脈絡分析-2014 年

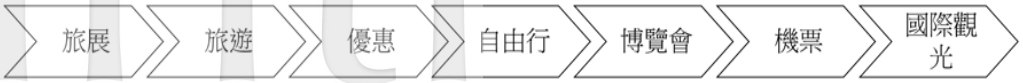

圖 4-1-6 脈絡分析-2015 年

(五) 情感分析

為了了解新聞文章內容是正面的還是負面的，利用情感分析去分析 2014 年與 2015 年夏季旅展的新聞文章內容。由圖 4-1-3 可知，在 2014 年的旅展新聞中，正面情感佔了 86%，而負面情感佔了 14%；由圖 4-1-3 可知，在 2015 年的旅展新聞中，正面情感佔了 93%，負面情感只佔了 7%；在這兩年之間的正面情感方面略有提昇。

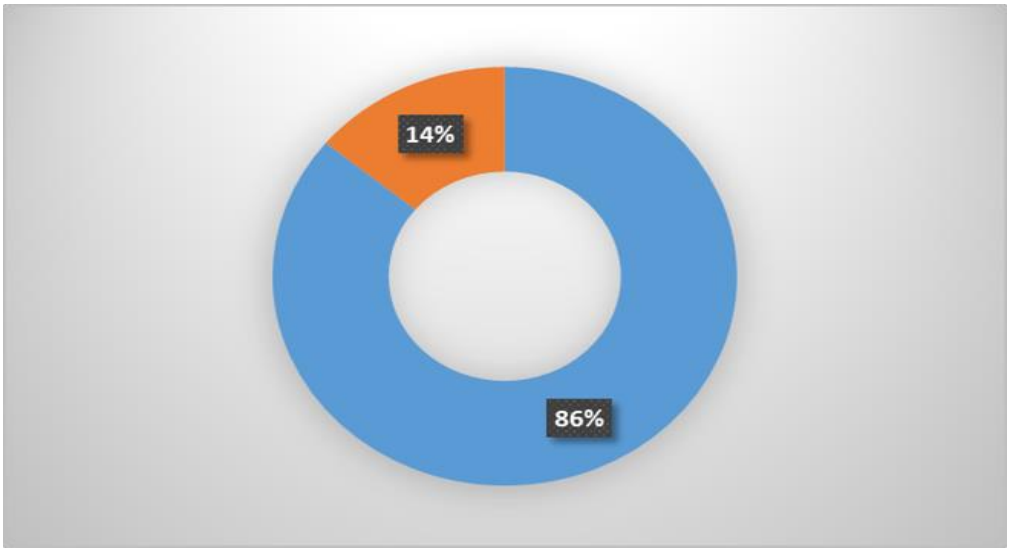

圖 4-1-7 情感分析-2014 年

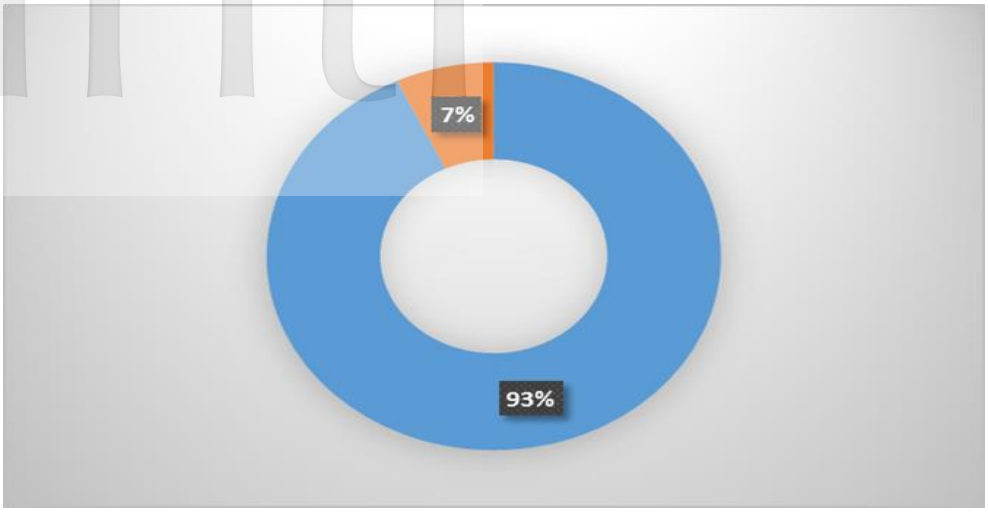

圖 4-1-8 情感分析-2015 年

二、案例結論

經由詞雲分析、關聯分析、集群分析、脈絡分析、情感分析，得出以下結果。

(一)夏季旅展在 2014 年與 2015 年的參展人數分別為 259,460 人與 282,865 人，是略為上昇的趨勢。

表 4-2-1 參展人數

	展覽時間	展覽地點	參展人數
2014	5/23-5/26	台北世貿一館	259,460
2015	5/22-5/25	台北世貿一館	282,865

(二)在詞雲分析中，夏季旅展這兩年都選擇了台北世貿中心作為展館，表示舉辦展覽的地點是重要的，可能會影響來參展的人數；在 2015 年出現了自由行、飯店、機票等字詞，可能表示民眾自由行的比例提昇，在這方面的配套行程可以在加強。

(三)在關聯分析中，2014 年與 2015 年與旅展相關的字詞第一名都是「優惠」，可見如有較多的優惠商品，可能吸引更多的參展人數。

(四)在集群分析中，2014 年優惠與飯店的關聯性高，飯店有沒有優惠可能會影響想來參展的意願；在 2015 年，國內、攤位、優惠為一群，可能表示國內的旅遊也是不可忽視的一群。

(五)在脈絡分析中，兩年之間的脈絡相差不大，2014 年主要是優惠與住宿，在 2015 年主要是優惠、自由行、機票，可以說明消費者對機票的價格比住宿更在乎。

(六)在情感分析中，大部分的夏季旅展新聞都是屬於正面情感，2015 年比 2014 年又略增加一些，表示正面情感的字詞越多，可以增加消費者前去參展的意願更高。

伍、結論與建議

文字探勘在近幾年月來越多人運用，不管是在網路上的資源，或是個人專屬部落格，都有不少人分享文字探勘技術的經驗，但是並不是普遍的人都懂得語法，所以建立此文字探勘平台，透過此平台就能夠簡易的去操作一系列的分析，讓使用者也可以輕易的把玩文字。

一、研究結論

本研究的文字探勘平台在架構上是以 Django 為網頁框架基礎，以 Python 為運算後台進行文字分析；Django 本身就是以 Python 的語法寫成，在 Python 上是最多人使用的網頁框架，所以與 Python 語法的相容性較高，平台的處理速度也算是差強人意，透過 Django 可以直接讓此平台變成一個簡易的文字探勘伺服器。

本研究平台提供了五種文字探勘的分析方法，讓使用者在大量的文字當中快速的挖掘出有效資訊，運用詞雲分析，可以很快的找出關鍵的字詞，運用關聯分析，可以得知什麼樣的字詞與討論的主題之間的相關程度，運用及群分析，將挖掘出的關鍵字詞做分類，運用脈絡分析，可以得知，文章或是網頁新聞等的重要脈絡，運用情感分析，能夠了解到討論內容字詞的正負面關係，最後運用視覺化呈現結果。

二、研究建議

本研究平台在在呈現方面並不夠完善，在前段的部分，需利用 Javascript 結合 CSS 做出動態網頁介面，讓介面可以看起來更加友善；在後台部分，可以隨著演算法的演進在做更新，讓中文字的分析能夠更加的準確；在資料呈現方面，可以加上 D3、ECharts 等視覺化的呈現效果，可使此平台更加的完善。

參考文獻

一、書籍及期刊

中文部份

1. 巫啟台(2002)。文件之關連資訊萃取及其概念圖自動建構。國立成功大學資訊工程學系所碩士論文。
2. 吳宜隆(2010)。建構於雲端運算之文字探勘服務系統。虎尾科技大學資訊管理研究所碩士論文。
3. 林名彥(2015)。應用文字探勘技術於客訴資料之研究-以台大 PPT 論壇為例。龍華科技大學資訊管理系碩士班論文。
4. 張文瑜(2005)。傳播學的建構-談問卷資料為什麼和如何被視覺化。中國廣告期刊，10 期，57-70。
5. 陳譽晏(2015)。運用 R Shiny 建立文字探勘平台之語意分析及輿情分析。輔仁大學統計資訊學系應用統計碩士班論文。
6. 陳柏江(2014)。運用文字探勘與推薦系統之技術建置失智症患者照護導引平台。國立臺北護理健康大學資訊管理研究所碩士論文。
7. 陳芸芸(2004)。視覺文化導論。台北，韋伯文化。
8. 鄭凱文(2014)。運用文字探勘及財務資料探討中國市場營運概況文字敘述及財務表現之一致性。國立政治大學會計研究所論文。
9. 劉育華(2014)。從文字探勘觀點分析臉書訊息－以台灣民間信仰的兩間宮廟為例。明新科技大學資訊管理系碩士班論文。
10. 謝邦昌、蘇志雄、鄭宇庭(2011)。SQL Server 2008 R2 資料採礦與商業智慧。臺北市，碁峰資訊股份有限公司。
11. 謝邦昌、鄭宇庭、李御璽、郭良芬(2011)。商業資料採礦 使用 Excel 2010。新北市，中華資料採礦協會。
12. 諶家蘭(2006)。淺介資料探勘與 XBRL。會計研究月刊，第 245 期：56-63。
13. 楊尊宇(2015)。基於 PCA、LDA 和 ICA 的資料視覺化研究。國立清華大學資訊系統與應用研究所論文。

英文部份

1. Bird, Steven, Ewan Klein, Edward Loper(2009), Natural Language Processing with Python, O'Reilly Media.

2. Fayyad, U., Piatetsky-Shaprio,G & Smyth, P.(1996).For, Data Mining to Knowledge Discovery :An overview .In advances in Knowledge Discovery and data Mining, 471-493.
3. Messaris, P.(1994). Visual literacy: Images, mind and reality. Boulder, Colorado: Westview Press.
4. Sullivan, A. (2001). Cultural capital and educational attainment. Sociology, 35(04), 893-912.
5. Thelwall, M., Wilkinson, D., & Uppal, S. (2010). Data mining emotion in social network communication: Gender differences in MySpace. Journal of the American Society for Information Science and Technology, 61(1), 190-199.
6. Tsytsarau, M., & Palpanas, T. (2012). Survey on mining subjective data on the web. Data Mining and Knowledge Discovery, 24(3), 478-514.
7. Van Rossum, G., & Drake Jr, F. L. (1995). Python reference manual. Amsterdam: Centrum voor Wiskunde en Informatica.

二、網路資料

中文部份

1. 資料視覺化網站
   http://blog.infographics.tw/2015/06/three-keys-to-visualization/
2. Python 第一次用就上手
   http://wiki.python.org.tw/Python/%E7%AC%AC%E4%B8%80%E6%AC%A1%E7%94%A8%E5%B0%B1%E4%B8%8A%E6%89%8B
3. Python 程式語言教學誌
   http://pydoing.blogspot.tw/2012/10/python-tutorial.html
4. Django Book 2.0
   http://docs.30c.org/djangobook2/index.html
5. Django Girls 指南
   https://djangogirlstaipei.gitbooks.io/django-girls-taipei-tutorial/content/index.html

英文部份

1. Python
   https://www.python.org/

2. PyCharm-JetBrains
https://www.jetbrains.com/pycharm/
3. Django
http://www.openfoundry.org/tw/tech-column/1330-django-