

國中歷史測驗題目自動分析系統

目錄

(一) 摘要	2
(二) 研究問題與背景	2
(三) 相關文獻探討	4
(四) 研究方法與步驟	5
(五) 研究結果	8
(六) 討論	15
(七) 後續研究與實作方向	20
(八) 參考文獻	21

(一) 摘要

本計畫研究目的是建構一個國中歷史科測驗題目自動分析系統，提供自動分析題目相關章節、題目難易度等多種資訊的服務，幫助學生學習及老師教學。本研究搜集國中歷史課本、題目、學生測驗讀卡結果，作為系統的輸入來源。研究方法中採用了 nodejieba 作為斷詞工具，並運用 Latent Dirichlet allocation (LDA), Support vector machine (SVM) 以及 Word2Vector 等文字探勘方法技術。研究成果製作出可提供歷史科測驗選擇題自動分類的模組，並完成伺服器以及網站介面等系統架構方便使用者進行雲端計算。

(二) 研究問題與背景

在現今的教學中，學生除了可以應用眾多的媒體，如網路影片、題庫平台，來幫助自己更有效率的學習。老師也可以妥善的運用許多工具，來節省在教學上繁瑣的工作，如此才能有更好的教學品質，對學生做更好的導引。但是現實的狀況，這些工具依然不是如此的方便使用，很多也並沒有實際的為使用者客製化。

所以我希望以本身在學習的資訊工程技術，來為老師以及學生們解決問題。

因此，我回到了母校台北市立東湖國民中學，訪問老師在教學上的一些阻礙以及困難，也訪問了國中生在讀書的時候，有哪些工作，覺得希望能有更好的工具來給予輔助。再經過我的規劃之後，促成了這項系統研究。

本研究專注於如何自動關聯歷史詞彙的章節關係，達到能為每一題無論是新或舊的歷史考題，自動給予章節關係強度分類。將使用者分為教師和學生兩者討論，探討其所能做出的應用，為了減低老師和學生在教學與學習上的負擔。

經過訪談調查之後，針對老師和學生遇到的問題整理如下：

(1) 考題很難有效的做章節分佈評測

在學校中每一學期，通常就會有兩到三次的段考。段考和一般測驗不同的是，老師為求考題適合當時教學進度以及學生學習狀況，甚至為了和當時的時事結合，都會親自去出整份考卷，從引文到選項的設計，都由老師一手包辦。這是一項非常困難的工作，而在教學和進

度的壓力之下，老師通常沒有餘力再去針對每一題做更詳細的分析與歸納。

這樣，可能就會產生有關鍵的概念漏考，或是有些章節出現的太多，壓縮到其他的章節空間。這當然都是一種出題的方式，但是如果能有工具幫助老師更了解考卷的數據分析情況，勢必可以讓考試的效果更好，也幫助測驗後的再練習。

(2) 考題的引文製作耗時

一份好的考卷，老師也會希望讓學生從考題之中可以將所學與現今的時事地理人文做結合。所以老師都會審慎的選擇考題的引文，不只在乎引文的正確性，也希望學生能從引文學到其他的延伸知識。但是引文的製作非常的困難耗時，希望能有工具針對題目或題組的內容，搜尋適合的相關引文。

(3) 考題的預期難度難以數據化

老師根據自身的教學經驗，和平時對於學生的觀察，都可以知道學生哪種題目比較不擅長，或是對哪部分的章節不甚熟悉。老師在出題的時候就可以大概評估此份考題學生的表現，適當的選擇應該給學生一些挫折與反省，或是使題目比較偏易一些來提起學生的學習興致。

但是這樣的經驗非常依賴於每位老師的個人經驗，如果能有工具幫助老師根據過去歷年來學生的作答狀況，去預測下次學生的表現，這對於新的老師會是很大的協助，有經驗的老師也可以有所參考。

(4) 學生很難針對不熟的題目加以複習

在練習的時候，海量的練習無論是書商提供的考卷或是參考書中的題目是必經的過程，但是很難針對錯過的範圍，直接快速的再找到相關的題目來做深度練習。

(三) 相關文獻探討

對於以自然語言描述的問題自動分析技術，以往的研究多考慮在CQA網站上，例如 Stackoverflow, Yahoo 知識+等開放式問答平台。

在2001年，Soubotin 等人[4]以及 Brill 等人[5]，在他們自己的 QA 系統中採用語意的分析、重組來表示問題的語意。而使用不同的語意模型，加入擷取的題目關鍵詞，就可以重新產生出各種闡述題目意義的語句。在這些成功的研究後，Ravichandran [7]、Kaisser [10]以及 Cucerzan 等學者[8]，也更進一步的發展出可自動學習的語句重組。

Riloff [9]研發了 Quarc系統，可以在閱讀一段英文文章後，從文字內容中找到語句回答固定類型的文章相關問題。Quarc 將五種典型的問題類別各自設計出規則集合，包括Who, What, When, Where, Why 等問句類型，這些問句類型與歷史科目題目中偏重人、地域、時間、文化、事件的描述重點極為相似，可以做為分析擷取題目類型及題目重點內容的參考透技術。

以準確性論，CQA 題目自動分類的準確性上因為描述領域過於龐大(如數學、社會學、人類學)而不易準確分類，這在限定考試科目的題目中是可以克服的。以必要性來說，教學測驗的題目屬性自動標示技術，不只可以方便出題時檢視考題性質的工作，更可以幫助教師和學生在考後檢視分析教學及學習時需加強的內涵。此技術也與一般要從自然語言描述句中轉換成關鍵字查詢，分析理解使用者查詢需求有關。

(四) 研究方法與步驟

(1) 確立問題，縮小問題範圍

在假設了一些研究問題之後，因為考量到研究時間，必須要將學科縮小為只有一門學科才行。國中主要和升學考試有關的科目：中文、英文、自然、公民、地理、歷史、數學，就屬歷史這門學科最為適合來做文字探勘、分類分析的研究。

原因在於歷史的專有名詞之間，可以建構出時間、地理、人文、事件等關聯性。我認為越多的關聯依據，有助於更準確的分類一個題目是和哪些章節比較有關係。再者，相較於數學自然科目，歷史以文字居多，比較不會有圖片、數學運算式等非純文字語意表達的因素出現，才比較不會分散了研究的核心。

(2) 教師訪談

為求更了解真實情況，我回到了自己的母校台北市立東湖國民中學，和歷史老師蔡美玲老師和張靜分老師聯絡，和他們解釋我想要研究的方向，並請教他們在教學上所面臨的一些困難。

(3) 學生訪談

除了了解老師的需求之外，我也希望能直接和在學的學生面對面，了解他們在學習上所遇到的困境。

雖然這不是一個營利的產品，但是這些訪談就像是市場調查，為了真正能讓研究出來的系統可以在教學上使用，直接去認識了解第一線的使用者的問題是我認為非常重要的。

(4) 資料蒐集

本研究所需要的幾項重要的資料，有課本內文、考卷、學生作答記錄，為了讓文字可以比較容易的給電腦程式來分析，在與學校老師訪談的過程中，和老師們取得了考卷的電子檔案，以及從學校教務處調出來的歷屆段考讀卡結果。不過因為系統有些異常，所以在讀卡結果的部分有所缺漏。

(5) 資料前處理

1. 轉為純文字

起初規劃使用圖文轉換軟體，將課本以及考卷的照片做文字轉換成純文字檔案，再撰寫程式來輸入演算法。但是經過實驗後發現，課本的插圖以及排版複雜，導致文字辨識非常不準確。因為這些文字將會是演算法核心的輸入來源，任何一字的誤差都會嚴重影響結果，所

以能直接從老師提供的電子檔案讀取內文，是非常重要的步驟。

但是因為學校作業系統相對老舊，老師所使用的 Microsoft Office Word 所儲存的文件檔案皆為 .doc 副檔名，這表示這些文件檔案都是二進位的 XML 檔案 (Binary XML)，因為無法確切得知 MS Word 的位元格式，所以無法直接使用程式來讀取文件。

這邊比較理想的解法，我先將每份文件用較新版的 MS Office Word 打開，將每一份文件轉換為 .docx，也就是新的 Open XML 格式。使用 Open XML 的好處是，課本中的一些如大章節標題、小標題，都會有固定的撰寫格式，這時就可以使用程式直接抓取相同格式的內文，就可以快速的蒐集完所有的章節名稱。

2. 中文斷詞

斷詞方面，先後嘗試了中研院的中文斷詞系統，以及結巴 (Jieba) 斷詞系統，在(六)討論中更詳盡的比較。

3. 建立歷史辭典

先將歷史課本後面的附錄附有中英對照表、人物名稱的關鍵詞彙等內容，建立起歷史詞彙字典。原因有以下幾點：

- 斷詞字典中沒有某些課本中的歷史詞彙
- 異體字統一標準 (如：台灣、臺灣)
- 音譯詞彙統一標準 (如：狄雅仕、迪雅士)

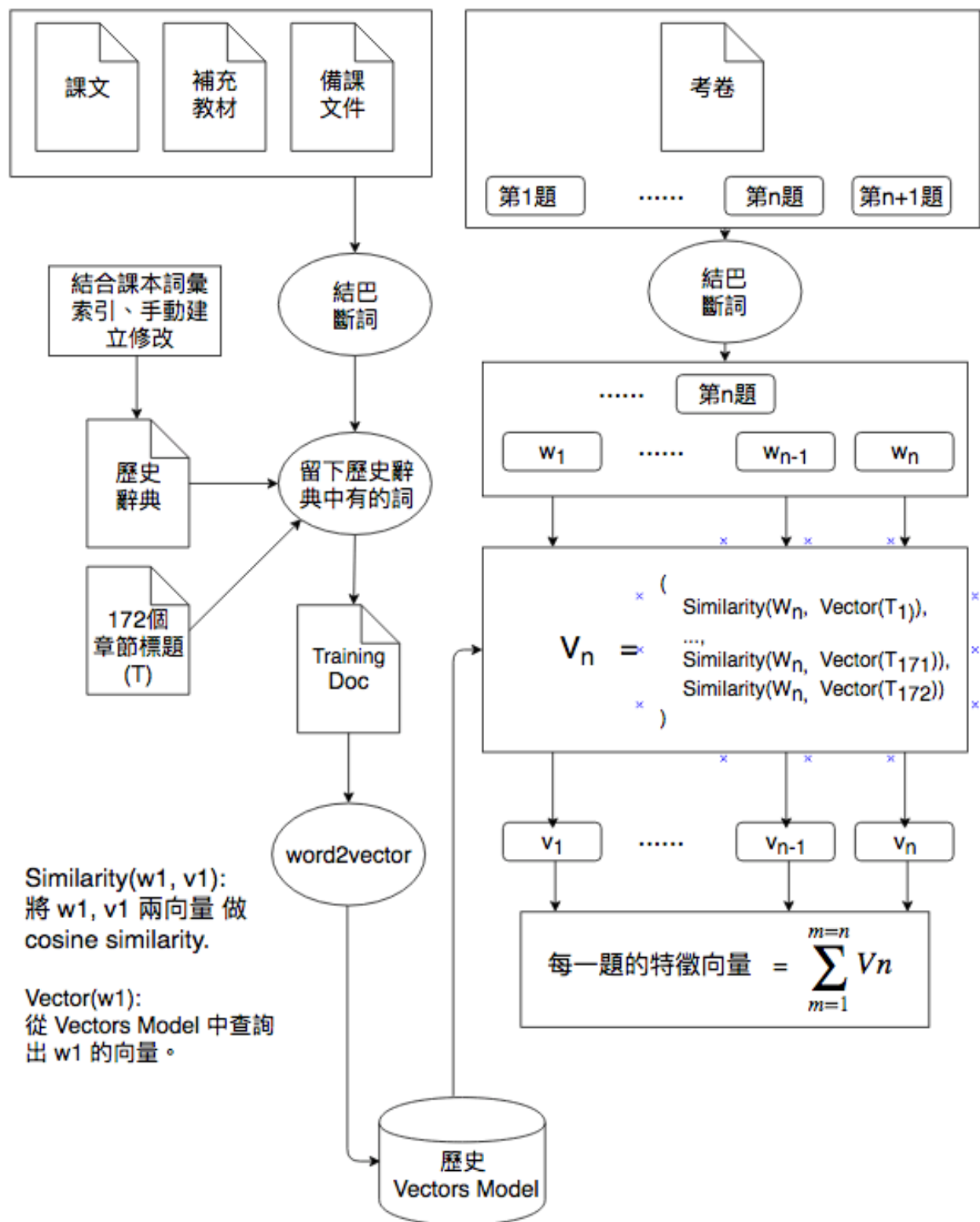
(6) 演算法修正

由於研究的方向有文本分類以及難度預測，所以在研究就過成中嘗試使用了 TF-IDF、LDA、Word2Vector 以及 SVM 等演算法。在經過比較之後採用了修正過後的 Word2Vector 搭配 SVM 來做為系統核心的算法。

(7) 伺服器、網站應用規劃

考量到使用者是老師以及學生，將系統建置為公開的網站，不只能跨作業系統，手機、電腦以及平板都可以使用。所以這個系統不能只是一個單機的工具，必須要能成為一個獨立運行的伺服器服務，並有網頁或是 App 等前端介面提供讓老師及學生操作試用。

(8) 系統核心規劃



(五) 研究結果

本研究使用了七份由東湖國中老師提供的段考考卷，共 280 題，皆為單選、無圖形作答的試題。

其中有：

- Single - Noun 188 題
- Multi - Noun 43 題
- Single - Phrase 40 題
- Multi - Phrase 9 題

輸入文本經過斷詞後，有 6945 個關鍵名詞詞彙。在使用 Word2Vector 時採用參數 windows 為 8，vector dimension 為 100 做計算。

(1) 題型分類

題目不考慮多選題以及帶有圖片的題目，在純文字化後，將所有的選擇題分為四類，原因有二：增加可能的難題屬性原因、正確區隔題目的主要關鍵內容。

1. 根據題幹有多選項與否，分成 Multi 和 Single 兩種子屬性。
2. 根據選項是否為專有名詞，分成 Phrase 和 Noun 兩種子屬性。

舉例：

Single - Noun

31. 葡萄牙人稱台灣為福爾摩沙，荷蘭人在台南建立熱蘭遮城，西班牙人在滬尾建立聖多明哥城，台灣歷史上所經歷的這些事，與下列哪一個事件相關？(A) 地理大發現 (B) 文藝復興 (C) 宗教改革 (D) 科學革命。

Single - Phrase

32. 十五、十六世紀的海外探險中，哪一項成就與西班牙無關？(A) 支持航海家探險，完成繞地球一周 (B) 在中南美洲取得金銀，成為歐洲最富強的國家 (C) 資助航海家，因而發現美洲 (D) 發現好望角。

Multi - Noun

33. 何者是十六世紀宗教改革後出現的新教教派？甲、希臘正教、乙、耶穌會、丙、英格蘭國教丁、路德教派。(A) 甲乙丙 (B) 乙丙丁 (C) 丙丁 (D) 乙。

Multi - Phrase

34. 地理大發現時期的海外拓殖，產生了哪些影響？甲、造成馬雅文化的消失、乙、改變美洲的人口結構、丙、傳染病傳入，造成美洲原住民大量死亡、丁、每週作物傳入歐、亞，增加糧食種類。(A) 甲乙丙丁 (B) 乙丙丁 (C) 甲乙丙 (D) 丙丁

如第 31 題，答案是地理大發現。如果希望能得到這一題的章節涵蓋範圍，意指非答案也必須被計算入內，則可以將四個選項經過 Word2Vector 訓練後的向量做計算，預期中若選項相近範圍的題目應當難度較高，反之選項明顯不同範圍的應該較易。

如果希望能得到正確答案的範圍，則只能將地理大發現納入此一題的屬性計算，其他三個選項的向量都不能納入考量。

(2) 相關詞查詢

透過 Word2Vector 輸入課文建立的向量模型，可以使用 cosine similarity 的排序，也就是夾角的大小排序，找到相近的詞彙。例如搜尋「孫悟空」，可以得到以下的結果

```
▼ [
  ▼ {
    "word": "唐三藏",
    "vector": 0.974369
  },
  ▼ {
    "word": "林黛玉",
    "vector": 0.951022
  },
  ▼ {
    "word": "神通廣大",
    "vector": 0.941816
  },
  ▼ {
    "word": "賈寶玉",
    "vector": 0.936503
  },
  ▼ {
    "word": "吳承恩",
    "vector": 0.933682
  },
]
```

中間雖然林黛玉和賈寶玉比作者吳承恩相關性還要高，一方面是因為它們在課本中都是分佈在小說選讀的附錄當中，一方面 0.02 的相似度差距相當微小，並不會影響在章節中分不相同的事實。

(3) 題目屬性數據化

為了可以使用 SVM 來做題目難度預測，必須先將題目的屬性量化成有代表性的數值，才能做出正確的預估計算。

以此題為例：

日耳曼人雖然被羅馬人視為蠻族，但在歷史的發展過程中，卻也有某方面的創新與進步，被歷史學者認為較羅馬帝國時代進步之處，應該是指何者？(A) 基督教的信仰 (B) 法蘭克王國的建立 (C) 農業生產技術的改良 (D) 中央集權的確立

我將屬性分成：長度、題型（如上所述）、完整斷詞、關鍵斷詞。其中關鍵斷詞的意義是將斷詞結果中，只留下我建立的歷史專有詞彙字典中的詞。目的是為了純化題目的主題，如範例的關鍵斷詞結果為：「日耳曼人、羅馬人、蠻族、羅馬帝國、基督教、法蘭克、農業生產、中央集權」。

但是這樣依然沒有得到可用的數值屬性，而 Word2Vector 的效果，正好可以應用在這個需求上。所以我將課文當作是訓練模型，讓課本中的每個詞彙都能擁有足以代表互相相關性的向量，如此就可以將關鍵斷詞的這些詞彙，轉換成向量的集合。

這時我們再將關鍵斷詞集與 172 個細部章節名稱做 cosine similarity，這一題就可以得到一個 172 維度的章節強度向量，當按照相關程度做排序之後，如下圖：

```
"章節關係-有排序": {  
  "西歐的封建社會": 0.6050108473023957,  
  "基督教的創立與發展": 0.588543416578328,  
  "拜占庭帝國": 0.47109053232903914,  
  "中世紀歐洲與伊斯蘭世界": 0.45630618523674726,  
  "西方文化的傳入": 0.4526663270984308,  
  "城邦分立的希臘文明": 0.36262009154238817,  
  "文藝復興": 0.35484435440441564,  
  "戰後臺灣的政治變遷": 0.34590351723766977,  
  "西周-封建的確立": 0.3436256920425091,  
  "歐洲的古典文明": 0.33582477729122245,  
  "蘇聯共產政權的發展": 0.33149144117099716,  
  "地理大發現與海外拓殖": 0.3269711907678009,
```

可以發現結果非常傾向於西歐的封建社會，也就是這個題目的來源範圍。

關於如何與章節做 cosine similarity，在 (六) 討論、Word2Vector 的使用技巧中有更詳盡的說明。

(4) 題目答對率預測

在上述的過程後，無論是舊的歷史題目或是使用者新出的歷史題目，都可以獲得一個歷史課本章節的分佈向量 (172 維度)。

這時候只要將每一個題目找到該次學生的答對率，就可以使用 SVM 來對其他的歷史題目做答對率的預測。SVM 的部分，因為答對率為連續的數值，所以採用了 Regression 的估計方式，並交互驗證答對率的 Pearson's correlation 來評測。

(5) 歷史模型查詢API製作

本專題最後的呈現，乃為架設了一台伺服器由本人撰寫了 API 接口來做試題分析服務。提供的功能有：

Word2Vector類型：

`/api/w2v/similarWords/:q/:n`

詞 q 相近的前 n 個字cos similarity

```
[
  {
    "word": "孫悟空",
    "cosineDistance": 0.9999999999999997
  },
  {
    "word": "沙悟淨",
    "cosineDistance": 0.9954645753570359
  },
  {
    "word": "唐僧",
    "cosineDistance": 0.992177600319716
  },
  {
    "word": "神通廣大",
    "cosineDistance": 0.9749238062658493
  },
  { ... }, // 2 items
  { ... }, // 2 items
  { ... }, // 2 items
  { ... }, // 2 items
  { ... }, // 2 items
  { ... }, // 2 items
  {
    "word": "唐三藏",
    "cosineDistance": 0.8858239500613585
  },
]
```

`/api/w2v/vector/:qs`

查詢 qs 中所有字(以逗點隔開)的向量

```
[
  {
    "word": "孫悟空",
    "vector": [
      -0.1777874908615408,
      -0.11858781674624333,
      -0.08123564141159136,
      0.02494123802084931,
      0.0711925539943892,
      -0.07460668971119526,
      -0.021524246720472824,
      -0.06224543955151807,
      0.07489110583481008,
      -0.07134675550719241,
      0.05673302102716008,
      0.10819577701673463,
      -0.027868211180540014,
      -0.04493489194757168,
      -0.0629987424973975,
      -0.08501986075912513,
      0.1466776212118487,
      0.15995437146420574,
      -0.204813304888804,
      -0.20436155156796196,
      -0.15310553982888656,
      -0.02347118359879197,
    ]
  },
]
```

History 類型：

`/api/history/analyze/:qString/:sort?/:detail?`

qString 為題目字串

sort 欄位(true, false)代表章節強度結果是否排序

detail 欄位(true, false)代表是否要印出每一個關鍵詞的章節強度結果

範例：發送 GET request：

`/api/history/analyze/日耳曼人雖然被羅馬人視為蠻族，但在歷史的發展過程中，卻也有某方面的創新與進`

`步，被歷史學者認為較羅馬帝國時代進步之處，應該是指何者？(A) 基督教的信仰 (B) 法蘭克王國的建立`

`(C) 農業生產技術的改良 (D) 中央集權的確立/true/false`

回傳的 JSON 物件有下列欄位：

```
▼ "完整斷詞": [
  ▼ {
    "word": "日耳曼人",
    "tag": "history"
  },
  ▼ {
    "word": "雖然",
    "tag": "c"
  },
  ▼ {
    "word": "被",
    "tag": "p"
  },
  ▼ {
    "word": "羅馬",
    "tag": "ns"
  },
  ▼ {
    "word": "人視",
    "tag": "x"
  },
],
"答對率": 37.04790186239862,
"章節關係-字": {
  "法蘭克": [],
  "基督教": [],
  "羅馬帝國": [],
  "羅馬": [],
  "日耳曼人": []
},
"章節關係": {
  "西歐的封建社會": 0.6050108473023957,
  "基督教的創立與發展": 0.588543416578328,
  "拜占庭帝國": 0.47109053232903914,
  "中世紀歐洲與伊斯蘭世界": 0.45630618523674726,
  "西方文化的傳入": 0.4526663270984308,
  "城邦分立的希臘文明": 0.36262009154238817,
  "文藝復興": 0.35484435440441564,
  "戰後臺灣的政治變遷": 0.34590351723766977,
  "西周-封建的確立": 0.3436256920425091,
  "歐洲的古典文明": 0.33582477729122245,
  "蘇聯共產政權的發展": 0.33149144117099716,
  "地理大發現與海外拓殖": 0.3269711907678009,
  "日治時期的經濟發展": 0.32269225986712397,
  "文化發展": 0.3175190511898288,
  "義大利的建國運動": 0.3168999664519183,
  "伊斯蘭世界": 0.3168517396634246,
  "二二八事件": 0.3079964438466978,
  "戰後臺灣的經濟與社會發展": 0.303046971854818,
  "羅馬的政治與文化": 0.29648243587219664,
  "牡丹社事件": 0.2939469555370475,
  "日治時期的社會與文化": 0.2903601667116642,
  "維也納會議後的歐洲": 0.28883435214553665,
  "民主化的歷程": 0.2847483934110307,
  "思想文化與科技": 0.2831073592861611,
```

此為完整的題目斷詞結果。

(6) 網頁應用系統

初版的分析系統於 history.speechlesslee.com 展示：

1. 輸入範例題目：使用者可以自行輸入欲查詢的題目，點下按鈕之後就可以得到運算後的結果。

試題分析

@Copyright 2016-2017 LeeXun

隱藏 範例 清空

*輸入題目，看看這題有多難

日耳曼人雖然被羅馬人視為蠻族，但在歷史的發展過程中，卻也有某方面的創新與進步，被歷史學者認為較羅馬帝國時代進步之處，應該是指何者？

*選項(A):

基督教的信仰

*選項(B):

法蘭克王國的建立

*選項(C):

農業生產技術的改良

*選項(D):

中央集權的確立

看看這題有多難？！

預估學生答對率：0%

章節相關強度：

2. 難度分析結果：以上題為例，可以發現分析出來的章節結果相當準確，這個功能主要是幫助老師在出題的時候，可以更容易檢視題目的章節分佈。

試題分析

@Copyright 2016-2017 LeeXun

+ 範例 清空

預估學生答對率：67.78%

章節相關強度：

西歐的封建社會:
0.8178220182870002

基督教的創立與發展:
0.7897975126487999


中世紀歐洲與伊斯蘭世界:
0.7657974497356

拜占庭帝國:
0.6956355962716001

義大利的建國運動:
0.6805082364582

歐洲的古典文明:
0.6220189380816

3. 相似題目查詢結果：學生可以利用此一功能，根據錯過的題目，找尋其他相關的題目做練習。



相似的題目：

1:阿俊上網查詢四世紀的羅馬帝國史事，不應該得到下列哪一項訊息？
(A)基督教合法化
(B)西羅馬帝國滅亡
(C)帝國分裂
(D)設立東都。7
因為題目與 羅馬帝國,基督教,羅馬帝國,東都,相關

2:日耳曼人雖然被羅馬人視為蠻族，但在歷史的發展過程中，卻也有某方面的創新與進步，被歷史學者認為較羅馬帝國時代進步之處，應該是指何者？
(A)基督教的信仰
(B)法蘭克王國的建立
(C)農業生產技術的改良
(D)中央集權的確立。2
因為題目與 日耳曼人,羅馬,羅馬帝國,基督教,法蘭克,相關

3:聖索菲亞大教堂經歷教堂變為清真寺的歷史，至二十世紀時，它以博物館的身份迎接來自世界各地的觀光客。在這棟建築物裡，基督教與伊斯蘭教的圖騰和平共存，這種難得的獨特現象，是在哪一種歷史背景下造成的？
(A)日耳曼人入侵
(B)阿拉伯帝國擴張
(C)商業復甦與城市興起
(D)拜占庭帝國滅亡。32
因為題目與 索菲亞,清真寺,二十世紀,觀光客,基督教,伊斯蘭教,日耳曼人,阿拉伯,拜占庭帝國,相關

(7) NPM Word2Vector 套件

將 word2vector 撰寫出優化的包裝供伺服器使用，釋出四個月後已經有許多下載量。

Overview

[train](#) [load](#) [getVector](#) [getVectors](#) [getSimilarWords](#) [getNeighbors](#) [similarity](#) [substract](#) [add](#)

w2v.train(trainFile, modelFile, options, callback)

[Click here to see example TrainFile format.](#)

Example:

```
var w2v = require("./lib");
var trainFile = "./data/train.data",
    modelFile = "./data/test.model.bin";
w2v.train(trainFile, modelFile, {
  cbow: 1,           // use the continuous bag of words model //
  size: 10,          // sets the size (dimension) of word vectors //
  window: 8,         // sets maximal skip length between words //
  binary: 1,         // save the resulting vectors in binary mode //
  negative: 25,       // number of negative examples; common value
```

Stats

80 downloads in the last day

249 downloads in the last week

304 downloads in the last month

No open issues on GitHub

No open pull requests on GitHub

Try it out

 [Test word2vector in your browser.](#)

Keywords

word2vector, word2vec, word, vec, machine learning, chinese, deep-learning, 向量, vector, 文字, 機器學習

詳見(<https://github.com/LeeXun/word2vector>)

(六) 研究結果討論

(1) .doc 與 .docx

大多數的 Microsoft Office 使用者，並不會去在意儲存的副檔名是哪一種，但是如此的代價就是若有天這些資料集需要做為資料分析的輸入源，在資料前處理就會有許多的不便。如：

- 無法正確的取得純文字內容
- 無法自由使用 xml 的格式來做特殊區塊的選取
- 使用非 Windows 作業系統時開啟檔案緩慢
- 一些如 Google 雲端等線上文件服務會無法打開過舊的檔案

(2) 考卷格式

```
var string = getExam( filename )
string.replace(全形符號).with(半形符號)
string.splitBy(題號)
string.splitBy(依序(A)(B)(C)(D))
```

除了將考卷的 .doc 檔案轉換成 .docx 的步驟繁瑣之外，目前對考卷做的前處理大致如上，在中文的考卷當中時常有半形全形交互使用的狀況，程式在閱讀的時候需要做些許的調整。

為了達到未來老師學生上傳考卷電子檔，就可以自動分析的需求，有一個統一的考卷格式，可以減少程式解析錯誤的情況。

(3) LDA 在歷史試題分析上的缺點

Latent Dirichlet allocation (LDA) 是常用來訓練主題模型，找出主題的關鍵字與主題分群。在研究中原本嘗試使用 LDA 來從文本中尋找出歷史的主題，希望能獲得一些特殊的結果，但是結果卻非常的糟糕，歸納原因如下：

1. 訓練文本的量太小

其實歷史課本中的每一個概念，敘述的文字通常都不會超過 70 字，課文只提供重點的敘述，供學生有第一步的理解。

2. 主題概念繁多複雜

準確來說，歷史課本就是一個在短小的篇幅中，包含台灣、中國、世界等歷史知識的文本。在這樣的情況下，LDA 非常難以準確地找到其中的主題。

3. 歷史名詞互相的關聯性複雜

雖然大致上分為台灣、中國、世界等三大範圍，但是其實這三者的關係緊緊相連。台灣史中期有西班牙人以及荷蘭人，其中鄭成功的歷史又會與中國歷史有所交疊。如此的例子不生枚舉，所以要 LDA 從中學習出準確的主題，非常的不適合。

4. 歷史課本本身就是一個主題分類完全的文件

因此，我才放棄使用 LDA，而改用了 word2vector。

(4) 將 word2vector 從創造屬性的工具，晉升為分類工具

在理解 word2vector 的原理後，我有了一個想法。若在產生訓練文件前斷詞的時候，我不要拘泥於原本的辭典，而是自己將課本的 172 個章節編輯成一個章節辭典，讓這些章節都成為一個 Word，這樣在 word2vector 的訓練之後，我就可以將任一個詞去對 172 個章節做 cosine similarity。

因為這些章節就是我希望取得的分類類別，在做完 cosine similarity 之後，就可以找到每一個詞的分類強度了。

而如此的分類方法，就非常的適合在歷史課本中這樣的文本環境：

- 本身已經有分類標題
- 文本內容多寡皆可
- 如新聞有標題 (Google News) 分類、雜誌有章節分類、小說有章節分類都非常適合

(5) 為何採用 Jieba (結巴)斷詞而非中研院斷詞系統

	Jieba 結巴斷詞	中研院斷詞
格式	JSON 小且不用 parse	Xml 大且 parse 過程複雜
完整性	無缺漏	回傳值常有缺漏
可用性	隨時可以使用	每日都會有關閉維護時段
離線	可以離線使用	需要線上環境
自定義辭典	有	無

自定義辭典這點尤其重要，因為我需要將 172 個章節自己定義成一個詞典，還需要將諸多的歷史名詞、異體字、簡繁體字等情況等考量，中研院斷詞無法滿足此實驗的需求。

	Nodejs	PHP
C/C++擴充	有原生的 API	有 PHP_CPP 擴充
文件完整性	較完整	幾無文件
套件成熟性	NPM 管理十分優秀	composer 上較無文字探勘的 套件
靜態檔案讀取	有原生讀取快取存放	需要配合 Apache，開發耗時
長久性的運算	性能相當優秀	不適合

(7) Node.js 優化

我採用 Node.js 最重要的原因在於，服務中有許多的 http request 背後都需要做 word2vector 以及 SVM 的運算，使用 JavaScript 以及 PHP 等動態語言速度上都會比較慢。而 Node.js 有提供 C/C++ addons 的 api，可以直接撰寫 C/C++ 並製作成 dynamic-link library 來讓 Node.js 讀取。

以下比較三種對 cosine similarity 的執行方式的速度：

執行環境 (macOS，處理器 2.7GHz Intel Core i5，記憶體 8G 1867 MHz DDR3)

項目	使用 Node.js 的 spawn 去執行一個已經編譯好的 getSimilarity.exe	將整個 vectors model 寫進 JavaScript 的變數當中，再操作變數計算 similarity	製作一個撰寫一個 getSimilarity 的 dll，再
執行 100 次 getSimilarity 的毫秒數 (ms)，取100次樣本平均	2801.198ms	0.163ms	0.014ms

可以看出，千萬不要經常使用 spawn 透過 Node.js 去執行一個二位元執行檔。因為兩者在 I/O 的資料傳輸會是非常大的速度瓶頸。

而雖然直接使用 JavaScript 變數來讀取 Model 的速度已經十分的快，但是在使用 word2vector 大多數的情況之中，vectors model 的大小都會超過 Node.js 給予 JavaScript 使用 Heap 大小上限的 1.9GB。所以想要避開這種狀況，使用 Node.js addons 的方法製作 dll，才是最好的方法，而且在速度上又比 JavaScript 的變數運算再快了十倍。

這樣的一個過程，大大的增加了伺服器的運算效率。[12]

(8) SVM 難度分析探討

為了評測使用 SVM 的難度分析結果，採用了以下兩種方式：

1. 將每一份題目 (40題) 分成兩份 (20題)。

訓練組：同一份考卷前 20 題 預測組：同一份考卷後 20 題

考卷編號	Gamma 值	Pearson's Correlation
1	0.5	0.01
2	0.05	0.31
3	0.15	0.26
4	0.2	-0.15
5	0.8	0.224
6	0.05	0.02
7	0.05	0.17

結論：每份試題各為40題，共280題。

Gamma 值眾數落在 0.05，Pearson's Correlation 平均為 0.2。

第一和第四份試題的Pearson's r會特別低的原因，從考題本身就可以發現，該份考卷幾乎沒有重複到章節的題目，導致學習的規則和所要預測的內容沒有關係，無法準確地預測出結果。

不過從其他份的考卷可以看出，學生的答對率和考題的章節、類型、長短是有0.2的正相關的。

2. 七份考卷，抽取其中一份做為預測組，其他六份為訓練組。

M：訓練的資料編號 N：測試的資料編號

訓練組編號	預測組編號	Gamma 值	Pearson's Correlation
2,3,4,5,6,7	1	0.05	0.41
1,3,4,5,6,7	2	0.65	0.47
1,2,4,5,6,7	3	0.1	0.23
1,2,3,4,6,7	4	0.4499999	0.089
1,2,3,4,6,7	5	0.05	0.36
1,2,3,4,5,7	6	0.05	0.28
1,2,3,4,5,6	7	0.15	0.399

```

{
  "train": "+./data/svm/2,3,4,5,6,7.svm.train",
  "test": "-./data/svm/1.svm.train",
  "m": "2,3,4,5,6,7",
  "n": "1",
  "bestGamma": 0.05,
  "bestCorrelation": 0.41038900286908336
}

```

第一組

```

{
  "train": "+./data/svm/1,3,4,5,6,7.svm.train",
  "test": "-./data/svm/2.svm.train",
  "m": "1,3,4,5,6,7",
  "n": "2",
  "bestGamma": 0.65,
  "bestCorrelation": 0.47042832169494536
}

```

第二組

```

{
  "train": "+./data/svm/1,2,4,5,6,7.svm.train",
  "test": "-./data/svm/3.svm.train",
  "m": "1,2,4,5,6,7",
  "n": "3",
  "bestGamma": 0.1,
  "bestCorrelation": 0.23482823300187658
}

```

第三組

```

{
  "train": "+./data/svm/1,2,3,5,6,7.svm.train",
  "test": "-./data/svm/4.svm.train",
  "m": "1,2,3,5,6,7",
  "n": "4",
  "bestGamma": 0.44999999999999996,
  "bestCorrelation": 0.08939558424131588
}

```

第四組

```

{
  "train": "+./data/svm/1,2,3,4,6,7.svm.train",
  "test": "-./data/svm/5.svm.train",
  "m": "1,2,3,4,6,7",
  "n": "5",
  "bestGamma": 0.05,
  "bestCorrelation": 0.36495439334375557
}

```

第五組

```

{
  "train": "+./data/svm/1,2,3,4,5,7.svm.train",
  "test": "-./data/svm/6.svm.train",
  "m": "1,2,3,4,5,7",
  "n": "6",
  "bestGamma": 0.05,
  "bestCorrelation": 0.2877209135448375
}

```

第六組

```

{
  "train": "+./data/svm/1,2,3,4,5,6.svm.train",
  "test": "-./data/svm/7.svm.train",
  "m": "1,2,3,4,5,6",
  "n": "7",
  "bestGamma": 0.15000000000000002,
  "bestCorrelation": 0.3997536887047694
}

```

第七組

從上面可以發現，訓練資料增加之後，預測的結果也提升了許多，而第四份考卷是屬於偏差比較大的考卷，但是相關係數也還是增加了。

結論：增加訓練的考卷數量，可以讓難度預測的準確度有顯著的提升，所以在往後的研究，會先以擴充輸入資料為優先。

(七) 後續研究與實作方向

(1) 擴充資料來源

目前演算法、資料前處理、系統架構基礎已經完備，最缺乏的部分就是大量的歷史考卷，以及其對應的考試統計結果。這方面非常需要學校的協助，而且雖然大多數的學校都使用讀卡器作為段考時評量的器材。但是設備不一定有專人在維護，所以歷屆的測試資料難以獲取。因此，希望能獲取更多計畫執行上的資助，與更多的學校合作，讓分析系統能更加的完善。

(2) 實際參與考試

目前分析評測部分都還是採用交叉驗證，未來計劃能直接和校方合作，配合學校的段考，來觀察系統預測的準確性。

(3) 學生系統使用

請學生在考試前使用分析系統做輔助學習，了解學生的學習狀況是否有獲得改善。

(4) 老師系統使用

提供免費線上環境讓老師可以在出題目時使用，了解系統可以再改進的地方並加以修正，以及新增可以提供的其他功能。

(5) 擴充涵蓋科目

在相關領域的 Conference 上分享成果，並招集有意願一起協作的夥伴，一起開拓其他的科目研究。其中還需要圖形辨識、教育理論、相關學科的專業領域人士的支援。更可以結合台灣師範大學他系的學生以及老師一起執行研究，為台灣更好的教育學習環境努力。

(6) 自動答題系統

目前對於題目的分類已經相當的準確，可以嘗試往機器人自動作答的方向研究，訓練機器可以回答歷史的問題。

(7) 與政府、書商或是補習班合作

不希望此一研究只是在大專專題之後就結束，而是能繼續找到有力的支持者合作。我認為在地化的台灣教育資源，這些課本和老師的題目絕對是 Google 等大廠所無法觸及的資源。唯有台灣人自己先開始運用資料研究的技術來實現下一個階段的教育方法，一起改變更多教育上的問題，才能讓台灣的教育環境更好。

(八) 參考文獻

- [1] <http://exam.naer.edu.tw/index.php#ResultList>
- [2] http://ckipsvr.iis.sinica.edu.tw/papers/category_list.doc
- [3] M. Linares-Vasquez, C. McMillan, D. Poshyvanyk, and M. Grechanik, “On using machine learning to automatically classify software applications into domain categories,” in Empirical Software Engineering, pages 7 – 8, 2009. Springer.
- [4] Soubotin, M. M., and Soubotin, S. M. 2001, Patterns of Potential, “Answer Expressions as Clues to the Right Answers,” in Proceeding of the TREC-10 Conference, NIST, pp175--182.
- [5] Brill, E., Lin, J. J., Banko, M., Dumais, S. T., and Ng, A. Y. 2001, “Data-Intensive Question Answering,” in Proceedings of the 10th Text Retrieval Conference (TREC-10), pp. 183--189.
- [6] L. Sehgal, N. Mohan, and P. S. Sandhu, “Quality prediction of function based software using decision tree approach,” in ICCMT, pages 43 – 44, 2012.
- [7] Deepak Ravichandran , Eduard Hovy, “Learning surface text patterns for a Question Answering system,” in Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, July 07-12, 2002, Philadelphia, Pennsylvania [doi>10.3115/1073083.1073092].
- [8] Cucerzan, S., and Agichtein, “Factoid Question Answering over Unstructured and Structured Web Content,” in Proceedings of the 14th Text REtrieval Conference (TREC 2005).
- [9] Ellen Riloff , Michael Thelen, “A rule-based question answering system for reading comprehension tests,” in Proceedings of the 2000 ANLP/NAACL Workshop on Reading comprehension tests as evaluation for computer-based language understanding systems, p. 13-19, May 04-04, 2000, Seattle, Washington [doi>10.3115/1117595.1117598]
- [10] Kaisser, M., and Becker, “Question answering by searching large corpora with linguistic methods,” in Proceedings of the 13th Text REtrieval Conference (TREC 2004).
- [11] Sanjay K. Dwivedi and Vaishali Singh, “Integrated Question Classification based on Rules and Pattern Matching” , in Proceedings of ICICS ’ 14, November 14-16 2014, Udaipur, Rajasthan, India.
- [12] <https://medium.com/developers-writing/how-to-get-a-performance-boost-using-node-js-native-addons-fd3a24719c85#.1qguryj4m>