國立政治大學經濟學系研究所

碩士學位論文

基於人力銀行之台灣地區薪資預測模型

Web-Recruitment Data for Salary Prediction in Taiwan

指導教授: 陳樹衡博士

研究生: 廖宜川 撰

中華民國一○九年七月

DOI:10.6814/NCCU202001406

摘要

本文的研究目的在於建構一個薪資預測模型，在此特別針對資訊軟體系統類相關職缺。此薪資預測模型可作為求職者與企業方的參考依據，根據結構化變數，包括個人資料與職位相關技能等等，以及工作內容的文字描述，可以讓他們了解該職位的大略薪資，減少雙方對於薪資的歧見。同時，從迴歸模型輸出的係數也可以知道各種變數所反映的市場價值，例如熟悉某項工作技能會對於薪資水準有甚麼樣的影響，提供求職者自我精進的方向與參考。本研究從資料的探索性分析開始，了解各個變數的基本特徵，並嘗試整合結構化變數(職位需求的條件等等)以及非結構化的變數(工作內容的文字描述)，藉由許多的機器學習演算法建立薪資預測模型。另外，也嘗試使用詞向量轉換的神經網路模型，針對工作內容的文字描述建立薪資預測模型，其評估結果並不亞於使用結構化變數的薪資預測模型，這顯示了中文的自然語言處理，應用於網路人力銀行資料集的薪資預測模型之建構是可行的。

關鍵詞: 薪資預測、機器學習、卷積神經網路、自然語言處理、Word2Vec、詞向量、高維數據

DOI:10.6814/NCCU202001406

Abstract

The purpose of this thesis is to construct a salary prediction model, especially for information software system related positions using web-recruitment data. Based on structured data, including personal information and job-related skills, as well as unstructured text describing job content, the established models can be used as a reference for job seekers and companies to estimate the salary level of a certain job. Meanwhile, the variable coefficients from the regression models provide information about the market value reflected by those variables. The identified high-pay skills and expertise could guide the job seekers in which areas they can improve themselves. This research starts with an exploratory data analysis which helps us to understand the basic characteristics of each variable. Next, we apply various machine learning algorithms to the integrated structured and unstructured data to establish salary prediction models. The results show Random Forest, Ridge and Lasso perform well on the sparse high-dimension dataset. After that, we adopt a natural language processing approach by employing a convolutional neural network on the word vector data transformed from job content text. The result shows that the created salary prediction model is on a par with the models constructed using integrated structured and unstructured data. This endorses natural language processing as a viable approach to construct salary prediction models using online recruitment data.

Keywords: salary prediction, machine learning, convolutional neural network, natural language processing, Word2Vec, word vector, high dimension data

DOI:10.6814/NCCU202001406

目次

DOI:10.6814/NCCU202001406

表次

DOI:10.6814/NCCU202001406

圖次

DOI:10.6814/NCCU202001406

DOI:10.6814/NCCU202001406

第一章 緒論

第一節 研究緣起與目的

求職是多數人必經的過程，作為工作的主要報酬，薪資是工作媒合過程中最重要攸關資訊，它直接影響求職者的工作意願與企業方的人力成本支出。了解職場上的相關能力、變數可以如何轉換為可量化的薪資，是求職者跟企業方都想要知道的問題，對於雙方而言，尋找一個合理的薪資能讓工作媒合的過程更加順利，同時，也能降低人力資源錯誤配置的可能性。

隨著網路發展日益普及，台灣地區民眾使用網路平台作為求職途徑已成為主流，根據創市際 2016 年的問卷研究即顯示，近七成的台灣民眾使用網路平台(包括人力銀行與社群平台)，作為取得工作相關資訊的主要媒介[1]，因此相對於其他媒介較具一般性。同時，網路平台的資訊可以迅速地以特定技術進行清理與整合，產生巨量資料加以分析，且其變數具有結構化與一致性，相較於傳統媒介是更適合的研究資料標的。

本文的研究目的在於建構一個薪資預測模型，在此特別針對資訊軟體系統類相關職缺，根據國內[2]與國外[3]的人力銀行研究報告都顯示，因應近年來的產業發展趨勢，這類職缺是求職者的熱門選項。同時，將研究範疇限縮在特定職缺類別，可以有效降低擅長工具、工作技能、職位等欄位的維度，避免資料矩陣過於稀疏，有效增加預測效能。但也因為限縮研究範疇，所以模型的解釋能力僅止於資訊軟體系統類相關職缺，但是在本研究中所使用的資料前處理、模型建構等研究方法，仍可以適用於使用其他職缺資料時的實務研究。

---

[1] 就業調查與就業服務/職涯類別網域使用概況 https://www.ixresearch.com/wp-content/uploads/report/InsightXplorer%20Biweekly%20Report_20160815.pdf

[2] AI 大浪捲動企業搶才職缺是 5 年前的 3.2 倍 https://corp.104.com.tw/archive/files/news/20200121.pdf

[3] These 5 high-paying, growing jobs didn't exist a decade ago—but they'll be booming through the 2020s https://www.cnbc.com/2019/12/30/5-high-paying-growing-jobs-that-will-be-booming-through-the-2020s.html?fbclid=IwAR1mOcFVDUNxaGk5EAsbkxLU2wP40yxLb8cBqNGjrccXgXoCoiuR4_LxTTQ

DOI:10.6814/NCCU202001406

此模型可作為求職者與企業方的參考依據，根據結構化變數，包括個人資料與職位相關技能等等，以及工作內容的文字描述，了解該職位的大略薪資，減少雙方對於薪資的歧見。同時，從迴歸模型的係數也可以知道各種變數所反映的市場價值，例如熟悉某項工作技能會對於薪資水準有甚麼樣的影響，提供求職者自我精進的方向與參考。

第二節 研究貢獻

本文使用台灣地區人力銀行[4]的資料集，整合結構化變數(職位需求的條件等等)以及非結構化的變數(工作內容的文字描述)，並藉由許多的機器學習演算法建立薪資預測模型，在此將本文的研究貢獻羅列如下:

1. 交叉比較不同迴歸模型與不同變數組合搭配下，在薪資預測模型資料集的預測效能，並依此建構薪資預測模型，作為求職者與企業方的參考依據。由於類別型與文本資料集的特性，本研究的資料集較為稀疏且高維，這對於建構薪資預測模型而言是一項挑戰。儘管如此，在所有變數組合上，隨機森林(Random forest)、Ridge 和 Lasso 始終優於其他演算法。而在這些演算法當中，實證分析結果發現，隨機森林迴歸模型在使用結構化的變數搭配詞頻的狀況下，能夠達到最低的平均絕對誤差(Mean Absolute Error, MAE)。在測試集的平均絕對誤差可以達到 8659 元，在該測試集的平均薪資為 47346 元的情況下，測試集的平均絕對誤差比例為 18.3%。

2. 透過徵才案件中變數之係數，了解各項變數之市場價值，作為求職者與企業方的參考依據。像是工作地區位於國外的工作案件，例如東南亞、日本，薪資水準都較台灣本地高出許多。另外，經歷、需負擔管理責任與長時間出差的工作案件，也對於

[4] 104 人力銀行 https://www.104.com.tw/jobs/main/

DOI:10.6814/NCCU202001406

薪資預測模型都具有相當顯著之影響。同時，我們也能夠從擅長工具、技能、職位的子表中了解哪些擅長工具、技能、職位具有比較高的市場價值，例如 Spring Framework 的知識、軟體品質與保證、軟體專案主管等等會有較高的薪資。

3. 在台灣地區，並沒有相關文獻直接使用工作內容的文字描述，作為薪資模型建構的變數。本研究第一次嘗試將工作內容的文字描述，使用 Keras 嵌入層自訓練詞向量，並調整優化 TextCNN 架構以建構薪資預測模型。實證結果發現在測試資料的平均絕對誤差可以達到 8656 元，在該測試集的平均薪資為 47346 元的情況下，平均絕對誤差比例為 18.3%，這樣的研究結果顯示中文的自然語言處理，應用於網路人力銀行資料集的薪資預測模型之建構是可行的。

第三節 論文架構

本論文共分為六個章節，第一章探討本文的研究緣起與目的，以及本文的貢獻。第二章羅列與本研究相關之文獻回顧，包括台灣與國外地區薪資模型建構的相關文獻，以及藉由中文自然語言處理作為預測模型變數的相關文獻。第三章則羅列本研究所使用的相關實證分析方法簡介。第四章詳述資料蒐集過程、前處理，以及各變數的探索性分析。第五章為薪資預測模型建構的實證結果，包括一般的迴歸模型與詞向量轉換模型。第六章為結論，並對未來的相關研究提出建議。

DOI:10.6814/NCCU202001406

第二章 文獻回顧

本章節就本文所使用的相關研究架構與研究方法做文獻探討。第一節主要探討台灣地區薪資模型的相關文獻，從這些文獻中參考他們所使用預測模型與變數。第二節則主要探討國外地區薪資模型的相關文獻，同樣也是將重點放在他們所使用預測模型與變數。最後，第三節則羅列使用中文的自然語言處理作為預測模型變數的相關文獻。

第一節 台灣地區薪資模型

劉姿君(1993)使用行政院主計處民國 79 年的人力運用調查資料，建構多元迴歸模型，旨在探討教育投資對於薪資所得之影響。經實證結果發現，教育投資能顯著增加薪資所得，而若工作地點在北部地區，將會有更高的教育收益率。另外，該研究也發現工作經驗對於薪資所得有正向影響，但是其收益率隨著工作經驗年份增加而減少，而兩者在比較之下，工作經驗的影響較教育投資更為顯著。因此本文也使用教育投資與工作經驗作為變數，檢視其在資訊軟體系統類的工作職缺上是否有同樣的效果，差異在於該研究的教育投資將學歷轉換為受教育年數，從類別變數轉換成連續變數。

相似地，周宜滿(2004)使用行政院主計處民國 84 年與 91 年的人力運用調查資料，建構多元迴歸模型，旨在探討人力資本相關變數對於薪資所得的影響程度。實證結果發現教育投資與工作經驗均對薪資所得有正向影響，且由民國 84 年與 91 的資料比較可得知兩者影響均日趨重要，但與劉姿君(1993)不同的是，該研究發現教育投資對薪資所得的影響較為顯著。

莊惠婉(2010)使用行政院主計處民國 96 年的人力運用調查資料，樣本選取男性受雇員工，並分別運用最大概似法(Maximum likelihood estimation)與二階段有序機率選擇建構模型，想要瞭解在不同產業別、不同公司規模的情況下，影響員工薪資的因素有哪

DOI:10.6814/NCCU202001406

些。經實證結果發現，居住地區不管在任何情境下(使用模型、公司規模、產業)對於薪資都有非常顯著的影響，因此本文也使用地區作為變數，想要探討該變數在資訊軟體系統類的工作職缺上是否有同樣的效果。差異在於該研究僅粗略地分為台灣的南部與北部地區，而本文則以台灣的城市為單位並包括些許國外地區，作為切分地區的依據。

林鼎晃(2013)使用 94 學年度畢業後一年的追蹤調查資料，運用一般化迴歸模型，探討各科系畢業生初入職場的薪資水準。研究發現部分科系對於薪資水準有顯著影響，例如工業技術、電機工程均顯著與薪資水準呈正向關係，因此本研究也將科系納入變數考慮，想要探討該變數在資訊軟體系統類的工作職缺上是否有同樣的影響效果。

第二節 國外地區薪資模型

Singh(2016)使用 Aspiring Minds' Employability Outcomes 2015 (AMEO,2015)資料集，探討哪些因素決定了印度工程師的起薪，該文分別運用 Ridge、Lasso、支持向量迴歸(Support vector regression, SVR)建立模型。實證結果發現英文能力與科系對於薪資的影響都是顯著的，因此本文也將這些變數納入考慮，其中，該研究的英文能力直接使用受測者大學時期的測試成績，本文則以工作案件對於英文能力的要求作為替代。此外，該資料集與本研究的資料集最大的差異在於，AMEO 2015 是以求職者的角度建構的資料集，其變數主要聚焦在求職者的特徵，如過去的成績或是一些個人的基本特徵；而本研究的資料集則是以企業的角度出發，主要的變數為針對職位對於求職者提出的一些條件或要求等等。

Mart´ın et al.(2018)使用西班牙的資訊相關工作招募網站案件作為資料來源。在資料前處理部分，使用 Exact mutual information for feature selection

DOI:10.6814/NCCU202001406

(X-MIFS)作為變數挑選的依據，並嘗試運用分群演算法，切出多個案件分群建立多個模型。最後使用多種演算法如隨機森林、支持向量機(Support vector machine, SVM)、K 最近鄰居法(K nearest neighbor, KNN)等建構薪資分類模型，並比較分類器效能，實證結果發現工作經驗與職位對於薪資有顯著影響；相反地，教育程度對於薪資水準的影響則不這麼顯著。本文仿照該研究之模型選擇模式，挑選出合適的薪資預測模型，且該研究與本文預測效能最佳的模型都是隨機森林，而差異在於本文未將薪資轉換成類別變數使用分類模型預測，而是使用迴歸模型預測數值，因此輸出的結果較為精準。此外，因為樹狀演算法的單一變數結果較不易解釋，本文也仿照該研究使用正規化迴歸模型輸出係數，了解各變數與薪資的關係。同時，該研究並未使用工作內容的描述文字作為變數，因此本文認為也許這是一個可以增進預測效能的新作法。

第三節 中文自然語言處理用於預測模型變數之相關研究

曾厚強、洪孝宗、宋曜廷與陳柏琳(2016)的研究資料集選取 98 學年度臺灣三大出版社 1-12 年級審定版的國語科、社會科、自然科及體育與健康教育等四個領域的教科書共計 6230 篇作為資料集。他們首先使用 WECAn 將中文文本斷詞，再透過 Word2Vec 取得連續詞袋模型的詞向量對照表生成文字特徵。最後搭配深層類神經網路與支持向量機模型進行比較，旨在建構可讀性的分類模型，將該資料集的教科書依據年級做正確的分類。實證結果發現不論是在三種領域還是四種領域資料集的情況下，深層類神經網路的準確率都優於支持向量機。在本研究中也使用類似的程序做分析，差異在於斷詞工具的選擇方面，本研究選擇使用 jieba[5]而非 WECAn；同時，本研究也並無使用 Word2Vec 套件做詞向量的轉換，而是直接選擇使用 Keras 的嵌入層得到詞向量，並在研究過程中在嵌入層考慮使用中文維基的預訓練詞向量，期待能增進預測效能與訓練效率。最後，本研究處理的是迴歸問題而非分類問題。

[5] 結巴中文分詞 https://github.com/fxsjy/jieba

DOI:10.6814/NCCU202001406

江易麋(2018)使用 YAHOO 的網路新聞資料集，運用機器學習與深度學習的模型，旨在利用文字特徵正確地分類這些網路新聞的主題。首先，他們使用中央研究院的中文斷詞系統(CKIP)[6]進行斷詞，其中文字特徵轉換方法分別為詞頻-逆向檔案頻率(Term frequency–inverse document frequency, TF-IDF)與詞向量轉換(Word2Vec: Skip-gram)。接著他使用 TF-IDF 作為變數，透過 SVM 和單純貝氏(Naïve Bayes)建構模型。另外，也使用詞向量轉換後的變數透過不同的深度神經網路，例如卷積神經網路(Convolution neural network, CNN)、長短期記憶神經網路(Long short-term memory, LSTM)、雙向長短期記憶神經網路(Bi-directional long short-term memory, BiLSTM)建構模型。比較上述不同變數與模型的搭配之後，實證結果發現深度學習搭配詞向量轉換相較於其他組合都有較佳的正確率，因此本文也使用類似的架構處理工作內容的文字描述。本研究中，我們將 TF-IDF 與結構化的變數結合，使用許多線性與非線性的迴歸演算法建構薪資預測模型。而針對以詞向量表示的資料集，則使用以 CNN 為基礎的演算法建構薪資預測模型，並比較以上各個薪資預測模型的預測效能。

徐豪(2019)的研究資料來源為蘇良剛分享於 Github 的淘寶網商品評論語料，與本研究相同，首先使用 jieba 做為斷詞工具，再者，運用詞向量的轉換(Word2Vec: Skip-gram，透過中文維基百科的文本作詞向量的預訓練，將文字以向量的形式表示)與搭配主成份分析(Principal component analysis, PCA)降維，再利用深度學習的相關技術，目標是將這些評論依據正負評價做分類。實證結果發現 LSTM 相較於多層感知器(Multilayer perceptron, MLP)、SVM 有較高的精確度。

[6] 中央研究院的中文斷詞系統(CKIP) http://ckipsvr.iis.sinica.edu.tw/

DOI:10.6814/NCCU202001406

第三章 研究方法

本章節旨在介紹研究過程中使用的相關研究方法。第一節說明本研究使用的資料蒐集方法。第二節說明本研究中使用的統計檢定方法，包括卡方適合度檢定、D'Agostino 檢定等等。第三節說明在本研究中所使用的迴歸模型，例如線性迴歸、Lasso、支持向量迴歸、隨機森林迴歸等等。第四節則說明中文自然語言處理的部分，包括應用於中文斷詞的 jieba 套件，其相關理論與應用、文字的變數轉換型態:詞頻(term frequency, TF)與 TF-IDF，最後則說明在詞向量轉換模型所使用的 TextCNN 架構。

圖 3-1 大略描繪了本文的研究流程，首先，我們從台灣的人力銀行網站蒐集了資訊軟體系統類的工作案件。接著將工作案件內的工作內容描述的文字斷詞，轉換成 TF 與 TF-IDF 作為變數，並與工作案件中的結構化變數如工作地區、學歷等等結合，透過多種迴歸模型計算出薪資預測輸出。另一方面，也考慮透過神經網路的詞向量轉換模型，將斷詞後的工作內容的描述文字，藉由 Keras 的嵌入層轉換成詞向量矩陣，並使用 TextCNN 的模型架構，產出最後的薪資預測輸出。

DOI:10.6814/NCCU202001406

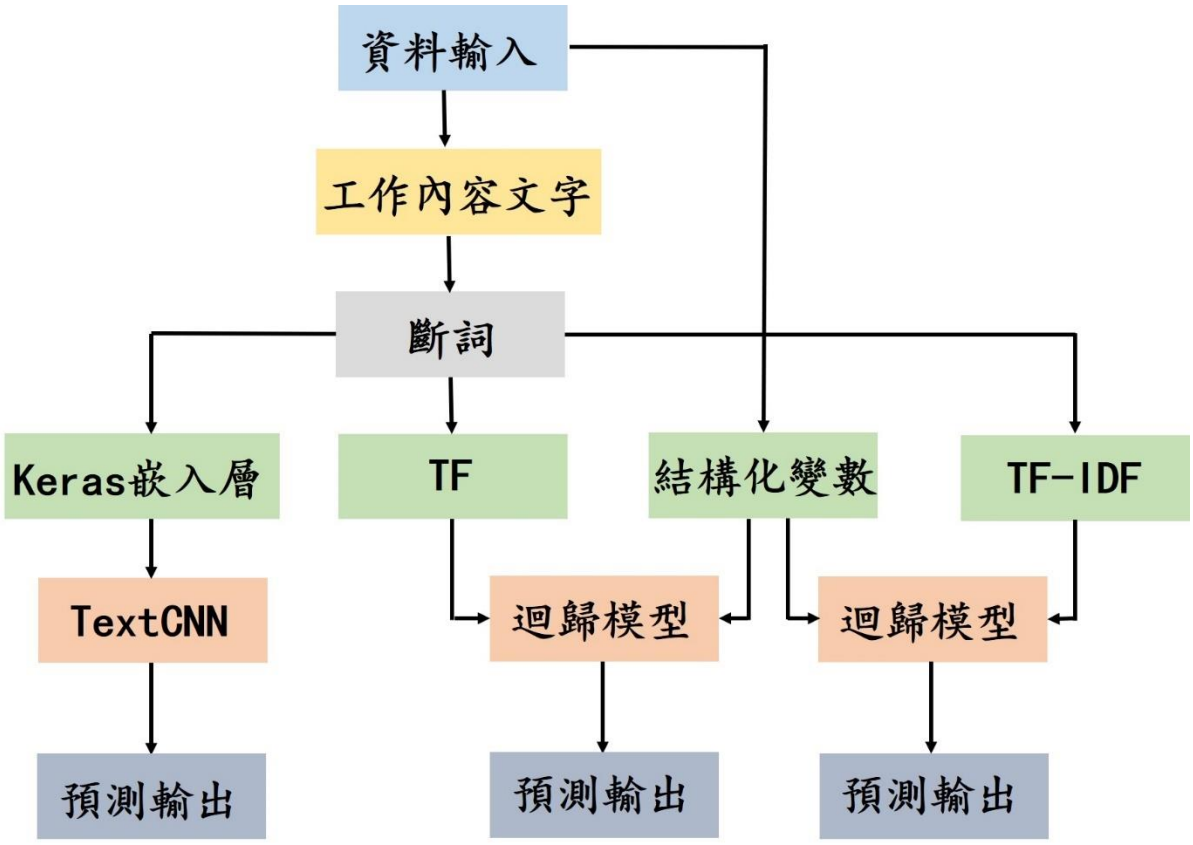

圖 3-1 研究架構圖

第一節 Selenium-WebDriver in Python [7]

Selenium WebDriver 是一個開源的爬蟲軟體套件，在 Python 的應用 API 特別熱門。為了使用 Selenium 爬取網頁資料，我們使用 Google Chrome 作為媒介，並手動選擇人力資源網站上的選項，以取得資訊軟體系統工作案件的網址，作為 Selenium 的入口網站。在實際抓取方面，本文是藉由抓取文字 XML 的位址，直接爬取網頁上的資訊。這些資訊包括工作案件的標題、工作地區、薪資、需要具備的技能等等，這些資訊將會再進行前處理，作為訓練薪資預測模型的變數。

7 Selenium with Python https://selenium-python.readthedocs.io/

DOI:10.6814/NCCU202001406

第二節 統計檢定

一、卡方適合度(Test of Goodness-of-Fit)檢定

卡方適合度檢定主要適用情況為檢視樣本資料分布與母體資料分布是否一致，檢定欄位只能是類別變項，虛無假設為兩資料分布一致，卡方統計值的計算公式如下。

$$\chi^2 = \sum_{i=1}^{N} \frac{(O_i - E_i)^2}{E_i} \quad (3.1)$$

其中O_i為樣本資料分布次數，E_i為母體資料分布次數，N為該變項的類別總數，自由度則為 *N-1*，若χ^2大於顯著水準的統計閥值，則拒絕虛無假設，此時代表樣本與母體分布不一致。

二、 D'Agostino 檢定

D'Agostino 檢定權衡資料之偏度與峰度以檢定資料分布，虛無假設為資料符合常態分布，統計值之計算公式如下。

$$\text{Statistics}=s^2+k^2 \quad (3.2)$$

其中 *s* 與 *k* 分別為偏度與峰度檢驗所得到的 z 分數值，若該 Statistics 大於某一顯著水準下的統計量閥值，則拒絕虛無假設，即資料集不符合常態分布。

三、 Mann-Whitney rank 檢定

Mann-Whitney rank 檢定主要適用情況為檢定兩母群體統計量(特別是中位數)的差異，且不需要假設母體為常態分布以及變異數相同。實際的計算方法是將兩樣本的資料

DOI:10.6814/NCCU202001406

做混合，依數值從小排到大去標註排序分數，再依兩樣本做分數加總，最後檢定兩樣本的分數輸出期望值的差異以推測兩母群體該統計量的差異。若其中一樣本分數期望值的 z 分數大於在某一顯著水準下的統計量閥值，則拒絕虛無假設，即兩母群體統計量明顯有差異。

第三節 迴歸模型

本研究中的迴歸模型均使用 Python 的 Scikit-learn[8]套件實現。在所有迴歸模型中都使用預設參數，這些參數設定在 Scikit-learn 的官方文件中都有詳細解說。

一、線性迴歸模型

3.3.1.1 線性迴歸[9]

線性迴歸旨在針對資料配適一個線性模型，藉由最佳化係數組合，極小化資料與預估值的殘差平方和，且在係數估計時要求變數之間是線性獨立的，否則會產生估計誤差，其目標式如下。

$$\min_{w} ||Xw - y||_2^2 \tag{3.3}$$

其中 X 為自變數矩陣，y 為應變數序列，w 為過程中需估計的係數。線性迴歸模型假設如下:

1. 隨機誤差項期望值為 0。
2. 隨機誤差項變異數一致。

[8] Scikit-learn https://scikit-learn.org/stable/
[9] Scikit-learn-LinearRegression https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html#sklearn.linear_model.LinearRegression/

DOI:10.6814/NCCU202001406

3. 隨機誤差項服從常態分配。

3.3.1.2 Lasso[10]

Lasso(Tibshirani, 1996)演算法在線性迴歸的基礎上，使用了 L1 正規化，使得部分估計的係數可能壓縮為 0，進而達到稀疏化和變數選擇的目的，能有效降低模型方差並防止過度擬合，該演算法也同時能處理共線性以及資料筆數少於變數數量的問題。其目標式如下。

$$\min_{w} \frac{1}{N} ||Xw - y||_2^2 + \lambda ||w||_1 \quad (3.4)$$

其中 X 為自變數矩陣，y 為應變數序列，w 為過程中需估計的係數，λ為係數的懲罰項，$||w||_1$則代表使用估計係數的絕對值總和，作為懲罰的目標項。在本研究中最佳的λ值利用網格搜尋，在 5 則的交叉驗證中尋找。

3.3.1.3 Ridge[11]

Ridge 算法在線性迴歸的基礎上，使用了 L2 正規化，使得部分估計的係數可能壓縮接近 0 但不為 0，因此仍在模型中保留所有變數，能有效降低模型方差並防止過度擬合，該演算法也同時能處理共線性以及資料筆數少於變數數量的問題。其目標式如下。

$$\min_{w} \frac{1}{N} ||Xw - y||_2^2 + \lambda ||w||_2 \quad (3.5)$$

[10] Scikit-learn-Lasso https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html#sklearn.linear_model.Lasso
[11] Scikit-learn-Ridge https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html#sklearn.linear_model.Ridge

DOI:10.6814/NCCU202001406

其中 X 為自變數矩陣，y 為應變數序列，w 為過程中需估計的係數，λ為係數的懲罰項，$||w||_2$則代表使用估計係數的平方總和，作為懲罰的目標項。在本研究中最佳的λ值利用網格搜尋，在 5 則的交叉驗證中尋找。

3.3.1.4 Kernel Ridge[12]

在 Ridge 的基礎上應用核轉換技術，基於不同的核轉換可以產生對應於原始空間的非線性轉換，期待能較 Ridge 有更佳的預測效能，常見的核轉換包括 Polynomial 和 Radial Basis Function(RBF)。其目標式如下。

$$\min_{w} \frac{1}{N} ||\phi_x w - y||_2^2 + \lambda ||w||_2 \quad (3.6)$$

其中ϕ_x為經由核轉換函數轉換後的自變數矩陣，y 為應變數序列，w 為過程中需估計的係數，λ為係數的懲罰項。

在本研究中嘗試在 Kernel Ridge 使用 RBF 核轉換，其定義如下式所示。

$$k(x_i, x_j)=\exp(-\gamma||x_i - x_j||^2) \quad (3.7)$$

其中x_i與x_j均為自變數的向量輸入，而$\gamma = \frac{1}{2\sigma^2}$，其中$\sigma$為一自由參數，其大小決定了模型的複雜程度。

3.3.1.5 ElasticNet[13]

[12] Scikit-learn-Kernel_Ridge https://scikit-learn.org/stable/modules/generated/sklearn.kernel_ridge.KernelRidge.html

[13] Scikit-learn-ElasticNet https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.ElasticNet.html#sklearn.linear_model.ElasticNet

DOI:10.6814/NCCU202001406

ElasticNet(Zou and Hastie, 2005)則介於 Lasso 與 Ridge 之間，可以使用超參數調整使用 L1 與 L2 正規化的比例。其目標式如下。

$$\min_{w} \frac{1}{2N} ||Xw - y||_2^2 + \lambda\beta||w||_1 + \frac{\lambda(1-\beta)}{2}||w||_2 \quad (3.8)$$

其中 X 為自變數矩陣，y 為應變數序列，w 為過程中需估計的係數，λ為係數的懲罰項，β則為調整使用 L1 與 L2 正規化比例的參數，預設為 0.5。在本研究中最佳的λ值利用網格搜尋，在 5 則的交叉驗證中尋找。

二、非線性迴歸模型

3.3.2.1 支持向量迴歸[14]

支持向量迴歸(Drucker et al., 1997)擴展支持向量分類(Support Vector Classification, SVC) (Vapnik, 1995)的方法解決迴歸問題，SVC的目標是找出一個使資料一分為二的超平面，以極大化與最近樣本的距離，而支持向量迴歸則是尋找一個能最佳配適資料的超平面如圖3-2所示。

[14] Scikit-learn-SVR https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html

DOI:10.6814/NCCU202001406

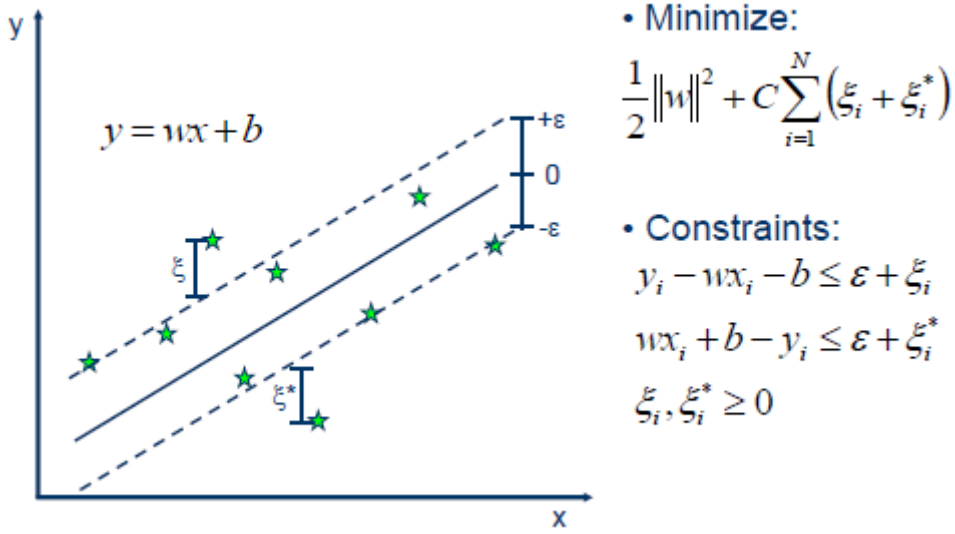

圖 3-2 支持向量迴歸示意圖[15]

過程中可以使用核轉換技術轉換資料維度，其目標式函數如下。

$$\min_{w} \frac{1}{2}||w||^2 + C\sum_{i=1}^{k}(\xi i + \xi_i^*)$$

$$s.t. \quad -\varepsilon-\xi_i \le w \cdot x_i + b - y_i \le \varepsilon + \xi_i^* \ \text{ and } \ \xi_i, \xi_i^* \ge 0 \qquad (3.9)$$

其中 x_i 為自變數，y_i 為應變數，k 為資料總筆數，w 為過程中需估計的係數，C 為係數的懲罰項，用來控制產生誤差時的懲罰係數，ε 為誤差區間，當迴歸的預測值與實際值超過 ε 的時候，需要給予一定的處罰，而 ξ_i, ξ_i^* 則為針對個別樣本的鬆弛變數，作為實際被懲罰的部分。

其主要優勢為:

1. 在高維空間的運算能力。
2. 根據核轉換指定不同的核功能。

[15] Support Vector Machine - Regression(SVR) http://www.saedsayad.com/support_vector_machine_reg.htm

DOI:10.6814/NCCU202001406

本研究在支持向量迴歸嘗試使用 RBF 核轉換，其定義如下式所示。

$$k(x_i, x_j)=\exp(-\gamma||x_i - x_j||^2) \quad (3.10)$$

其中x_i與x_j均為自變數的向量輸入，而$\gamma = \frac{1}{2\sigma^2}$，其中$\sigma$為一自由參數，其大小決定了模型的複雜程度。

3.3.2.2 多層感知器迴歸[16]

多層感知器(Hinton, 1989)由神經元連結所組成，其具有自我適應的學習功能，非線性的模型特色使得它能夠學習各種函數型態。其通常是由多個隱藏層所構成，每一層都會有輸入與輸出，基本的結構如圖 3-3 所示。

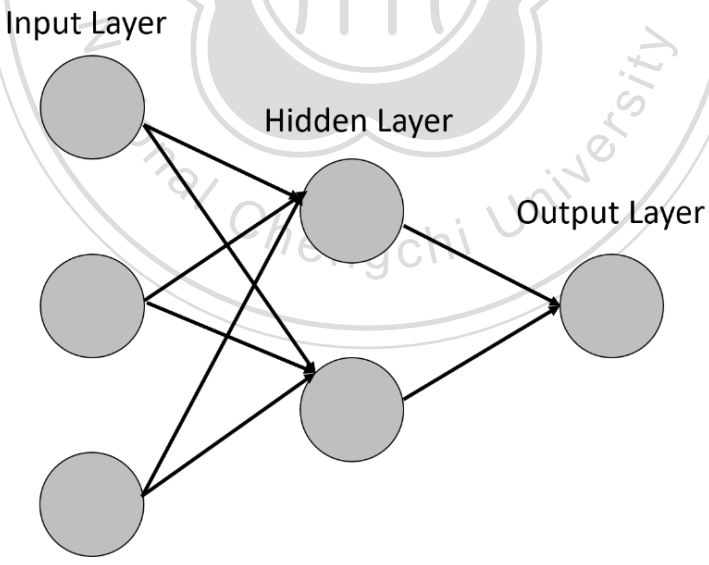

圖 3-3 多層感知器迴歸架構圖

[16] Scikit-learn-MLPRegressor https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html

DOI:10.6814/NCCU202001406

在每一個神經元中都有如圖 3-4 的結構，其中 l_1 到 l_n 為輸入向量，w_1 到 w_n 則分別是這些向量的權重，b 則是與輸入向量無關的偏差，N 則為輸入向量與權重相乘的總和，即 $N=l_1w_1+l_2w_2\ldots\ldots+b$，N 再透過激活函數 F 即可得到該神經元的輸出 output，其中激活函數的選擇往往與訓練的目標直接相關。

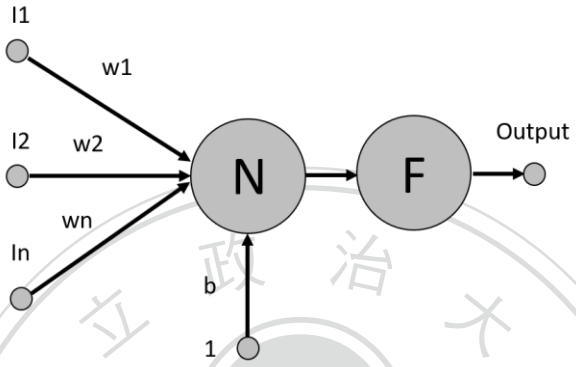

圖 3-4 神經元架構圖

其訓練的目標函數如式 3.11 所示，其中 X 為自變數矩陣，y 為應變數序列，w 為過程中需估計的係數。而係數則以隨機梯度下降(Stochastic gradient descent, SGD)演算法做更新，更新的模式如式 3.12 所示，其中W_t為目前的權重，W_{t+1}則為更新之後下一期的權重，l_r 為更新係數的學習率。SGD 透過隨機挑選樣本與變數數量以減少訓練時間，並且以不斷迭代的方式更新權重矩陣，直到收斂至一定的標準。該演算法使用倒傳遞(Backpropagation)的訓練方式更新權數，其基於現在的權重與偏差，向前傳遞計算出損失函數(Loss function)，接著，基於損失函數，倒傳遞利用損失函數梯度更新權重(包括偏誤)。在本研究中我們使用整流線性單位函數（Rectified Linear Unit, ReLU）作為輸出函數，並使用了一百個神經元的單一隱藏層。

$$L = \min_{w} \frac{1}{N} ||Xw - y||_2^2 \tag{3.11}$$

DOI:10.6814/NCCU202001406

$$W_{t+1} = W_t - lr\nabla L(W_t) \tag{3.12}$$

3.3.2.3 K 最近鄰居迴歸[17]

K 最近鄰居迴歸演算法(Altman, 1992)預測的輸出，是藉由找出訓練資料中與被預測資料最相似的 K 個資料集，在本研究中將 K 設為 5，取該資料群體的標籤平均。另外，也可以自行設定 K 個資料點的權重比例，以及計算最近鄰居的演算法，例如歐氏距離(Euclidian distance)或曼哈頓距離（Manhattan distance）。在本研究中所使用的為歐氏距離，其計算方法如下式。

$$\sqrt{\sum_{i=1}^{d}(x_{i,p} - x_{i,q})^2} \tag{3.13}$$

其中 d 表變數的數量，p 與 q 則代表兩者為任意不同的資料。

3.3.2.4 決策樹迴歸(Decision tree regression)[18]

決策樹迴歸(Breiman et al., 1984)依據簡單的決策規則為資料做分支，並使用遞迴或貪心演算法不斷增加分支。決策樹迴歸的預設分支標準為平均平方誤差在母節點與子節點的變化量，且每個節點上均有一預測值，作為截自該節點的預測輸出，最後的預測輸出則是該分支的所有母節點的數值平均。該演算法主要的優點有兩者，一是易於理解與解釋，再者是該模型對於噪聲有很好的穩健性，但是樹狀演算法有時難以處理過於稀疏的矩陣。

[17] Scikit-learn-KNeighborsRegressor https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html
[18] Scikit-learn-DecisionTreeRegressor https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html

DOI:10.6814/NCCU202001406

3.3.2.5 隨機森林迴歸[19]

隨機森林(Breiman, 2001)是一種元估計器，結合多個決策樹估計模型，再透過投票或是權重平均的方式決定最終輸出。預設使用決策樹作為單一估計器，其中各個決策樹為獨立的，且每個決策樹使用抽完放回的樣本抽樣模式，建立各個決策樹之間的差異。在模型建構過程中，可以自行選擇使用的決策樹數量(在本研究中樹的數量為 100 棵)，以及每棵決策樹中所使用的樣本數量(在本研究中每棵決策樹的訓練都使用全部的樣本)，在樣本的抽樣過程使用的統計方法為自助抽樣(Bootstrap)。另外，也可以選擇在分割節點的時候，設定所使用的最大變數數量(在本研究中每棵決策樹的節點分割都使用全部的變數)。這些隨機性的手段是要降低單一決策樹估計所產生的平均平方誤差。一般而言，單個決策樹通常表現出較高的方差且容易過擬合。在隨機森林中以隨機性的方式產生決策樹，藉由這些個別決策樹預測的平均值，可以有效地降低誤差。

3.3.2.6 AdaBoost[20]

AdaBoost(Freund and Schapire, 1997)與隨機森林迴歸相似，是一種元估計器，結合多個估計模型，再透過投票或是權重平均的方式決定最終輸出，預設使用決策樹作為單一估計器，但是各個子估計器並非獨立。在實際訓練時給予各個訓練樣本一個權數，在後一個估計器調整樣本權重的比例，調整依據是增加前一個估計器未正確預測的樣本權重，反之則降低，透過如此不斷地迭代產生最終模型。缺點是在模型建構過程中，無法選擇每棵決策樹中所使用的變數與樣本數量，無法藉由這兩個隨機性的手段降低單一決策樹估計所產生的方差。

第四節 中文自然語言處理

[19] Scikit-learn-RandomForestRegressor https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html
[20] Scikit-learn-AdaBoostRegressor https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostRegressor.html

DOI:10.6814/NCCU202001406

一、jieba(結巴)中文分詞[21]

中文文本處理很重要的一個步驟是分詞，因為中文不像英文可以直接使用空格做分詞，且各單字本身就有明確的意涵。中文字詞成詞的字元數量其實是不一定的，在成為字詞之後，中文字才會被賦予明確的意涵。所以在研究中文文本處理的時候，需要將句子作為輸入並以字詞作為單位輸出。

目前知名的中文斷詞系統分別為中研院開發的CKIP(Chinese Knowledge Information Processing)[22]和jieba。jieba是Sun Junyi(Sun, 2020)以Python撰寫的一個中文分詞開源工具，在本研究中使用jieba作為斷詞工具是因為其易於使用且同時有很大的修改彈性，能夠輕易地將jieba套件導入Python，並接續使用其它套件進行中文文本分析的研究。同時，其字詞庫與模型訓練日趨完善，許多中文自然語言處理研究都選擇其作為斷詞工具。jieba基於預訓練字典掃描文本，生成所有成詞情況的有向無環圖(DAG, directed acyclic graph)，並基於此利用動態規劃找出字詞切分的最大概率組合。而對於不存在於預訓練字典的新詞，則使用基於中文成詞的隱藏式馬可夫模型(Hidden Markov Model)，利用維特比(Viterbi)演算法求解，計算詞組的BMES(B:開頭，M:中間，E:結尾，S:詞)的最大組合概率。以下為兩個實際應用jieba精確模式的中文斷詞範例，其中使用"/"作為斷詞後的字詞區隔之符號。

一、"獨立音樂需要大家一起來推廣，歡迎加入我們的行列"

↓

獨立/音樂/需要/大家/一起/來/推廣/，/歡迎/加入/我們/的/行列

[21] 結巴中文分詞 https://github.com/fxsjy/jieba

[22] 中央研究院詞庫小組中文斷詞系統 http://ckipsvr.iis.sinica.edu.tw/

DOI:10.6814/NCCU202001406

二、“歡迎大家一起關注獨立音樂，讓大家能夠更認識獨立音樂”

↓

歡迎/大家/一起/關注/獨立/音樂/，/讓/大家/能夠/更/認識/獨立/音樂

二、詞頻(TF)與詞頻─逆向文件頻率(TF-IDF)

TF 與 TF-IDF(Sparck Jones, 1972)均可作為模型預測的輸入變數使用，兩者都可以直接從斷詞之後的文本計算得出。

TF 僅單純統計文本中所有出現過的字詞之頻率，欄位即為字詞，值則為該字詞出現在每則文本中的次數，其公式如式 3.14 所示。

$$TF(i,j) = \frac{n_{i,j}}{\sum_{k=1}^{N} E_{k,j}} \tag{3.14}$$

其中 j 代表的是某個特定文本，i 代表的則是文本中的某個特定字詞，$n_{i,j}$即為某個特定字詞出現在文本 j 的次數，$\sum_{k=1}^{N} E_{k,j}$則為該文本中所有字詞的數量總和，其中 N 代表的是文本 j 中共有多少個不同的字詞，$E_{k,j}$則代表了這些不同的字詞各自出現的次數，TF(i, j)將會介在 0 到 1 之間。

TF-IDF 則為詞頻的綜合加權，由兩部分組成，前半部分即為式 3.14 的 TF，其公式如下。

$$\mathrm{TF-IDF(i\,,j,D)} = \mathrm{TF(i,j)} \times \log\left(\frac{D}{1+\{j:t_i \in d_j\}}\right) \tag{3.15}$$

DOI:10.6814/NCCU202001406

D為所有文本的總數，$\{j: t_i \in d_j\}$為所有文本中包含某特定字詞t_i的文本數量。藉由上式可計算出特定字詞在特定文本中的頻率，並使用該字詞在所有文本中出現的頻率作為加權，以計算出該字詞在特定文本的重要性。相較於 TF 只考慮字詞出現在特定文本中的數量，TF-IDF 則更進一步地考慮到該字詞在所有文本中的重要性。其最後的輸出格式與詞頻相同，欄位即為字詞，值則為式 3.15 的計算結果。

TF 與 TF-IDF 的維度是文本中相異的字詞總數，因此當資料集很大時，資料維度也會擴張的很快，因此我們去除文本中特殊符號(例如”，”和”、”等等)，以縮小資料維度並去除噪聲，同時，由於英語單詞在大多數的工作案件中出現的頻率較低，為避免他們在計算 TF 和 TF-IDF 時有過多噪聲，我們從文本中刪除了 41 個停用詞（例如”of”和”with”）。表 3-1 與表 3-2 分別為使用第四節第一小節的中文斷詞結果所計算出來的 TF 與 TF-IDF 之範例。

表 3-1 TF 範例表

	一起	加入	大家	我們	推廣	歡迎	獨立	能夠	行列	認識	關注	需要	音樂
一	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0	0.1	0	0	0.1	0.1
二	0.091	0	0.182	0	0	0.091	0.182	0.091	0	0.091	0.091	0	0.182

表 3-2 TF-IDF 範例表

	一起	加入	大家	我們	推廣	歡迎	獨立	能夠	行列	認識	關注	需要	音樂
一	0.259	0.364	0.259	0.364	0.364	0.259	0.259	0	0.364	0	0	0.364	0.259
二	0.224	0	0.448	0	0	0.224	0.448	0.315	0	0.315	0.315	0	0.448

三、詞向量轉換神經網路架構

DOI:10.6814/NCCU202001406

本研究中的詞向量轉換神經網路架構首先將斷詞後的中文語料當作輸入，透過Keras[23]的神經網路嵌入層，將每個字詞轉換成多個維度的詞向量表示，最後使用TextCNN(Kim, 2014)的架構輸出預測數值。

3.4.3.1 Word2Vec

Word2Vec(Mikolov et al., 2013)是由 Google 開發的一個詞向量訓練模型的工具，其訓練的模式包括 CBOW 和 skip-gram，兩者訓練的主要差異如圖 3-5 所示。其中 CBOW 將某特定字詞的上下文當作輸入，訓練的目標即是預測該字詞；skip-gram 則相反，其以某特定字詞作為輸入，訓練的目標則是預測該字詞的上下文。兩者訓練模式均可藉由各自的訓練目標，產生一個可以將字詞轉換成以向量表示的權數矩陣。Word2Vec 訓練的過程通常以一個大型的文本語料庫為輸入，並產生一個具有數百個維度的向量空間，該語料庫中的每個字詞都會被分配到一個對應的向量，這樣的詞向量轉換方法可以使得意義相近的詞彙在向量空間中的距離較為靠近。詞向量的訓練模型通常是較淺的神經網絡，經過訓練可以建構單詞在向量空間中的關係。

[23] Keras: 基於 Tensorflow 使用 Python 撰寫的深度學習應用程式介面 https://keras.io/

DOI:10.6814/NCCU202001406

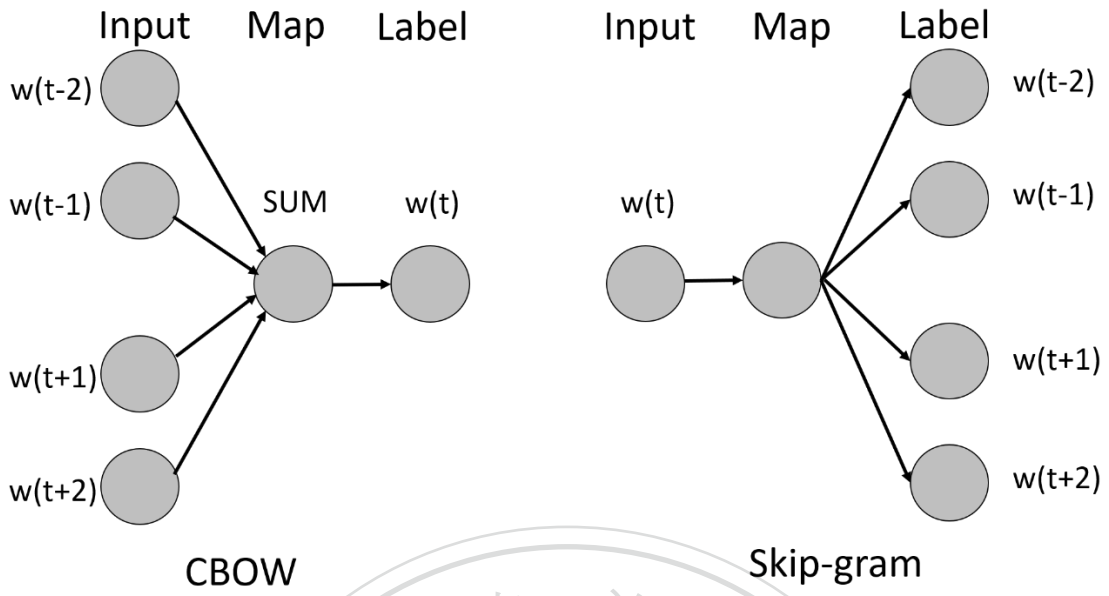

圖 3-5 Word2Vec 訓練模式示意圖

下表 3-3 為使用第四節第一小節中的中文斷詞結果所計算出來的權重矩陣，使用中文維基百科的預訓練模型，並透過 Word2Vec 的 skip-gram 做訓練。其中列代表的是每一個不同的輸入字詞，行則是該字詞的以向量表示的型態，該預訓練模型設定之輸出向量維度為 300，這裡只列出前 10 個維度呈現。

表 3-3 Word2Vec 權重矩陣範例表

字詞	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
獨立	0.14157	0.9014	-0.4579	0.19618	0.50296	-0.3086	-0.4156	-0.1917	0.14468	0.13502
音樂	-0.0051	0.74794	-0.3814	0.14345	0.3772	-0.578	-0.4693	-0.0014	0.40957	0.00256
大家	0.20885	0.72084	-0.4413	0.15866	0.90682	-0.2848	-0.6104	0.04185	0.13721	-0.4194
一起	-0.1066	0.69573	-0.48	0.42219	0.58003	-0.4926	-0.6651	-0.098	0.01529	-0.0524
歡迎	0.00737	0.93973	-0.5278	0.44187	0.5343	-0.512	-0.3367	-0.0026	-0.0441	0.07537
需要	-0.001	0.73851	-0.5607	0.38467	0.61987	-0.3357	-0.7487	-0.0602	0.10787	-0.0517
來	-0.013	1.4423	-0.8098	0.69711	0.78954	-0.563	-0.6419	-0.1605	0.16836	-0.1526
推廣	0.01409	0.74848	-0.6507	0.23621	0.19881	-0.5761	-0.4451	0.19082	0.12842	-0.2961
加入	-0.0457	0.85238	-0.3501	0.04789	0.21663	-0.3939	-0.3747	-0.0146	-0.0283	-0.2119
我們	0.13268	0.78522	-0.5	0.22848	0.60563	-0.4851	-0.6407	-0.1902	0.31774	-0.0258

DOI:10.6814/NCCU202001406

的	-0.1639	1.0639	-0.7227	0.38855	0.80669	-0.3977	-0.6485	-0.0203	0.32016	-0.0186
行列	-0.1026	0.62049	-0.6056	0.36945	0.42868	-0.2277	-0.3058	-0.0289	0.22526	-0.3084
關注	-0.2429	0.62925	-0.4733	0.42987	0.44391	-0.2366	-0.2926	-0.1021	0.07687	-0.0794
讓	0.07216	1.3592	-0.664	0.69052	0.69547	-0.5746	-0.7391	-0.058	0.35061	-0.2511
能夠	0.04764	0.91596	-0.4665	0.52974	0.48982	-0.4722	-0.5101	-0.1195	0.37495	0.09726
更	-0.185	1.2034	-0.5498	0.54276	0.80783	-0.3115	-0.5653	0.05028	0.10969	-0.3902
認識	0.10833	0.89034	-0.3296	0.36427	0.67561	-0.7105	-0.6097	-0.0855	0.27004	-0.1504

3.4.3.2 Keras 嵌入層

Keras 的神經網路嵌入層是將以稀疏矩陣表示的類別變項，映射到連續向量空間中的一個工具，訓練目標即是一個轉換矩陣，矩陣大小為字詞長度×詞向量維度，基於變數轉換的目的，嵌入層必須在神經網路架構的第一層。文字的嵌入是其一個熱門的應用，而關於詞向量生成的嵌入層訓練方式主要可分為三種。我們嘗試研究這三種不同的嵌入方式，其分別為 CNN-static, CNN-rand 和 CNN-non-static (Kim,2014)。

首先 CNN-static，其可以直接在嵌入層中使用預訓練的詞向量模型，作為權數矩陣的初始參數。本研究中所使用的預訓練詞向量模型之訓練文本為中文維基百科，並使用 Word2Vec 的 skip-gram 模式作訓練。使用預訓練的詞向量模型的結果當作初始權數矩陣，此時嵌入層的權重矩陣較能反映出字詞與字詞之間的真實關係，因為預訓練的文本通常較要測試的資料集大上許多。

再者 CNN-rand，則是直接使用整體神經網路模型的架構，依據其所對應的標籤，以反向傳播的模式訓練嵌入層中的權數矩陣，在這個情況下，嵌入層的初始權重矩陣是隨機生成的，並由標籤直接訓練詞向量，此時的主要目標較不是反映出字詞與字詞之間的真實關係，而是以訓練目標的損失函數作為主要考量。

DOI:10.6814/NCCU202001406

最後 CNN-non-static 則是前兩者嵌入層訓練方式之結合，以預訓練模型產生的初始權數矩陣為基礎，搭配整體的神經網路開始擬合訓練標籤，最後收斂的詞向量即是兩種目標之間的權衡，即在考量字詞與字詞之間的真實關係的同時，也將訓練目標的損失函數納入考慮做訓練。

3.4.3.3 TextCN 模型架構

TextCNN 是 Yoon Kim 在《Convolutional Neural Networks for Sentence Classification》(Kim, 2014)一文中首次發表的模型，其將卷積層的神經網路模型架構應用在詞向量文本的分類研究中，其模型架構如圖 3-6 所示，主要特色是使用了多個不同窗格大小並聯的卷積層，搭配池化層，最後使用全連接層產生預測輸出，其中，這裡卷積層的窗格大小指的是在卷積過程中，特徵轉換圖的大小為何，例如若窗格大小為 5，在卷積的過程中會將每五個字的詞向量視為一個單位，再透過過濾層(filter)將這五個字的詞向量轉換成一個新的特徵。卷積層的輸出是特徵圖，然後將其傳遞到最大池化層中，該池化層再將每個特徵轉換圖的最大值作為其輸出的特徵。這個方法是為了在每個特徵圖中得到最重要的特徵—也就是具有最大值的(Collobert et al., 2011)。最後，這些特徵被傳遞到完全連接層，其最後的輸出則是文本的分類，而我們將會使用這個模型架構訓練薪資預測模型。

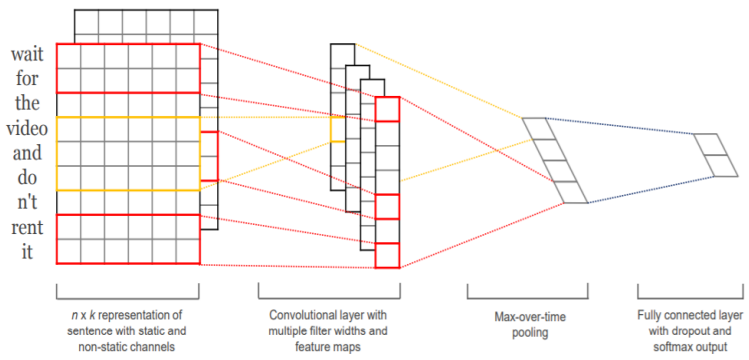

圖 3-6 TextCNN 架構示意圖 (Kim,2014)

DOI:10.6814/NCCU202001406

第四章 資料前處理與探索性分析

本章節旨在說明資料集的處理過程。第一節主要討論資料蒐集的過程，以及資料集的基本特徵。第二節則針對個別變數，進行變數的前處理與探索性分析，並特別著重其與薪資之關係。第三節特別處理工作內容的文字描述部分，將其轉換為模型可接受的變數形式。最後第四節則總結了經過本章前處理後的資料型態。

第一節 資料取得

於 2020/1/21 擷取 104 人力銀行網站上的徵才案件，使用 Python 的 Selenium 套件，搜尋 XML 的路徑結構，針對資訊軟體系統類職缺抓取 2935 筆案件。如圖 4-1 所示，每個案件都有固定的變數形式，包括職務類別、工作待遇等等。另外如圖 4-2 的範例所示，同時也會有非結構化的工作內容的文字描述，兩者都將會在後續做進一步的處理。

圖 4-1 結構化變數範例圖

DOI:10.6814/NCCU202001406

圖 4-2 工作內容文字範例圖

最後抓取的資料集共計 21 個變數，扣除掉 7 個與工作內容不直接相關的變數後，剩餘 14 個變數作為後續模型預測使用，欄位名稱如表 4-1 所示。可以從表中發現，除了薪資應變數為連續型資料，以及工作內容為文字以外，其餘變數均為類別型態的資料。相關的變數前處理，將會在下一節詳細討論。

表 4-1 變數欄位名稱與資料型態

變數名稱	資料型態
薪資	連續
地區	類別
管理責任	類別
出差要求	類別
上班時間	類別

DOI:10.6814/NCCU202001406

周休	類別
經歷	類別
學歷	類別
科系	類別
語言	類別
工作內容	文字
擅長工具	類別
工作技能	類別
職位	類別

以資料集的工作地區與台灣各縣市的工作人口進行卡方適合度檢定，台灣各縣市工作人口之資料來源為勞動部的勞動統計查詢網站[24]，結果為拒絕虛無假設，意即此資料樣本與台灣各縣市之工作人口母體分布不相符。可能的原因有兩者，一是使用網路徵才的族群地區分布本身即與此母體分布不同，再者是資訊軟體系統類的工作在各縣市之分布本身也不符合該母體分布。因此，本研究之解釋範疇僅止於網路人力銀行資訊軟體系統類的工作職缺，而非台灣所有的工作職缺。

第二節 資料前處理與探索性分析

一、薪資

薪資是本研究中的應變數，使用的單位為新台幣(NTD)。首先，排除任何不是以月薪作為薪資單位的案件，如年薪、日薪等，再者，將包含模糊意涵的薪資案件排除，如包含面議、以上等用詞，若薪資是以明確的範圍表示，則取其上下限的平均作為該工作案件的最後薪資，此時，總資料筆數即從 2935 筆變為 2633 筆。接著，針對薪資進行

[24] 勞動部勞動統計查詢網 https://statfy.mol.gov.tw/map01.aspx

DOI:10.6814/NCCU202001406

D'Agostino 常態分配檢定，發現薪資與對數薪資均不服從常態分配。在本研究中不對於薪資做轉換是因為想要保留原本薪資的單位，且目前很多的迴歸演算法也不特別要求應變數需要是常態分配。

薪資分布如表 4-2 與圖 4-3 所示，可以發現分布些微地右偏，且包含許多高薪資的離群值，且由表 4-2 可知資訊軟體系統類工作的薪資中位數為 42000。另外，該工作類別相較於同時期人力銀行網站上的所有工作案件共計 23439 筆，其薪資水準的分布較高，且標準差較大。

表 4-2 薪資敘述統計表

資料集	數量	平均值	標準差	第 1 四分位數	中位數	第 3 四分位數
資訊軟體系統類	2633	47310	18475	36000	42000	52500
所有工作	23439	36107	12820	29900	33500	38500

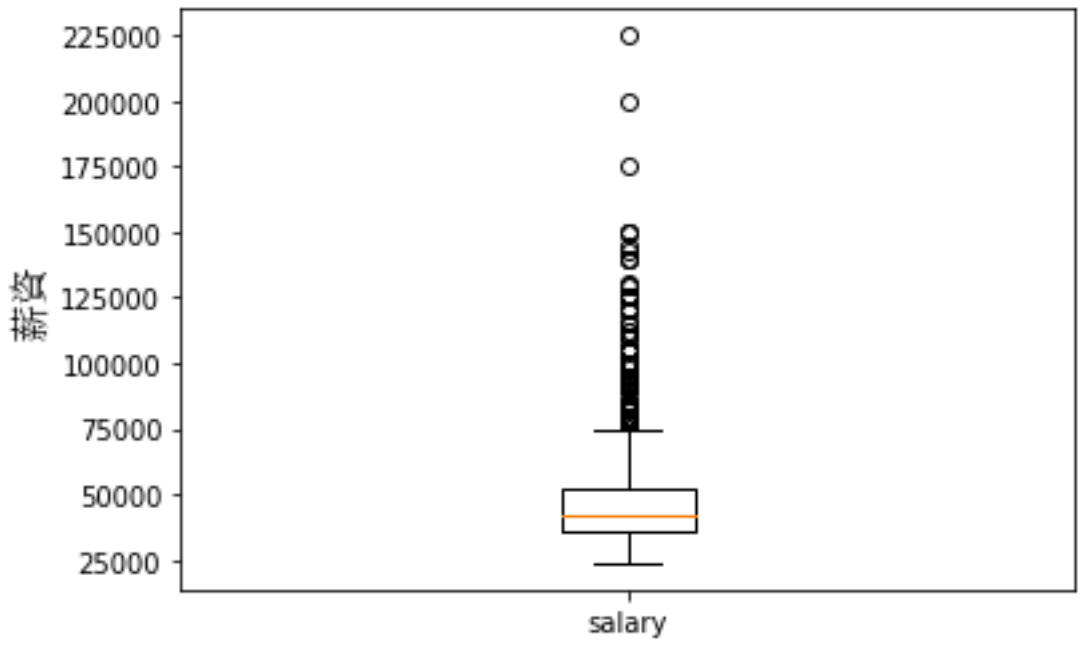

圖 4-3 資訊軟體系統類工作薪資分布箱型圖 (圖中橘線為中位數)

DOI:10.6814/NCCU202001406

二、地區

原先共有 1554 種不同的地址，在此以台灣的城市為單位整理，並將東南亞國家整合為單一類別[25]。實際操作如表 4-3 是擷取地址開頭兩字串做合併，經過此一操作，地區的個數縮減為 22 個。其中以位於台北之工作案件數量最多。

表 4-3 地區轉換範例表

轉換前地區	**轉換後地區**
台北市內湖區	台北
台北市大安區	台北
高雄市三民區	高雄
馬來西亞	東南亞

由圖 4-4 可以發現工作案件數量最多的地區位於台北，接著是台中與新北。在國外地區，因為地緣關係，因此工作案件也集中在亞洲地區，如東南亞、中國、日本等等。

[25] 這是由於單一的東南亞國家的數量過少，統計顯著性不足，可能會導致模型的嚴重偏誤，因此在此以地理位置的關係，將這些國家整合為單一類別。

DOI:10.6814/NCCU202001406

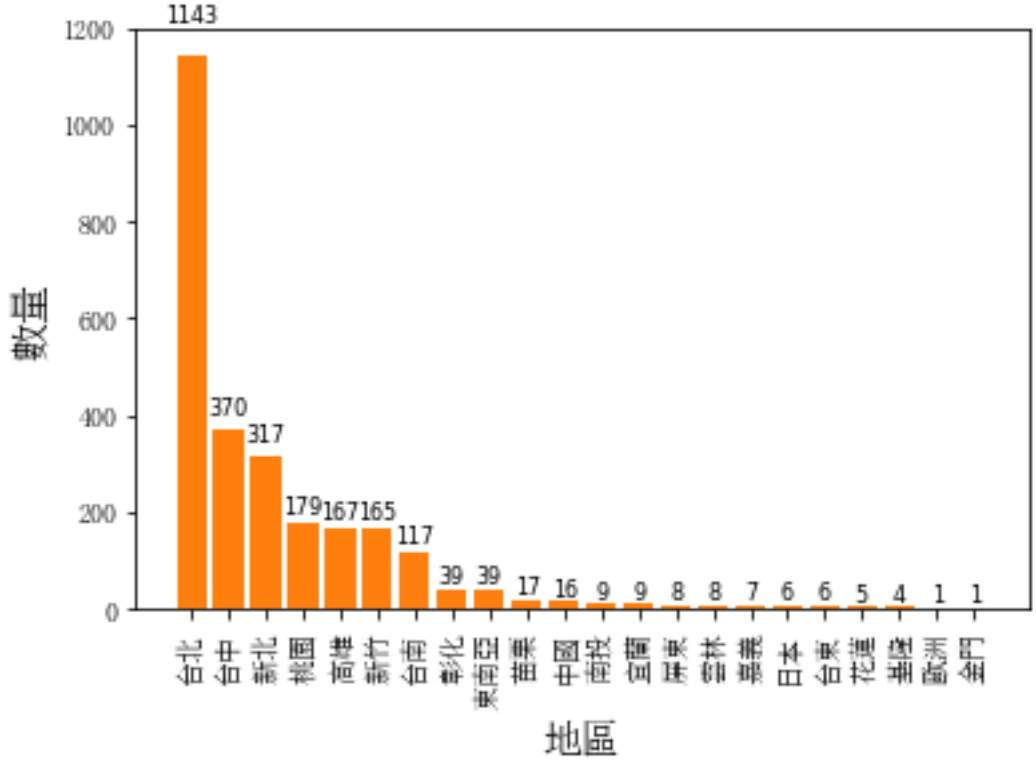

圖 4-4 地區分布柱狀圖

由圖 4-5 中可見台北地區較其他地區有較多的高薪機會且為右偏分布，甚至出現許多高薪的離群值。根據 Mann-Whitney rank test 的檢定結果(見附錄表一與表二)，在顯著水準為 0.01 的情況下，部分地區如新北與桃園的薪資中位數沒有差異，然而，其餘大部分在台灣的工作地區與台北的薪資中位數就顯著有差異。此外，國外地區如東南亞、日本、中國的工作薪資水準整體偏高，可見在台灣地區招募求職者至國外工作，需要以高薪作為補貼求職者前往國外工作的方式。這樣的結果與莊惠婉(2010)相似，該研究發現在所有類別的工作，工作地區對於薪資是有顯著影響的；而在本研究中也同樣發現工作地區也可以解釋資訊軟體系統類相關職缺的薪資差異。

DOI:10.6814/NCCU202001406

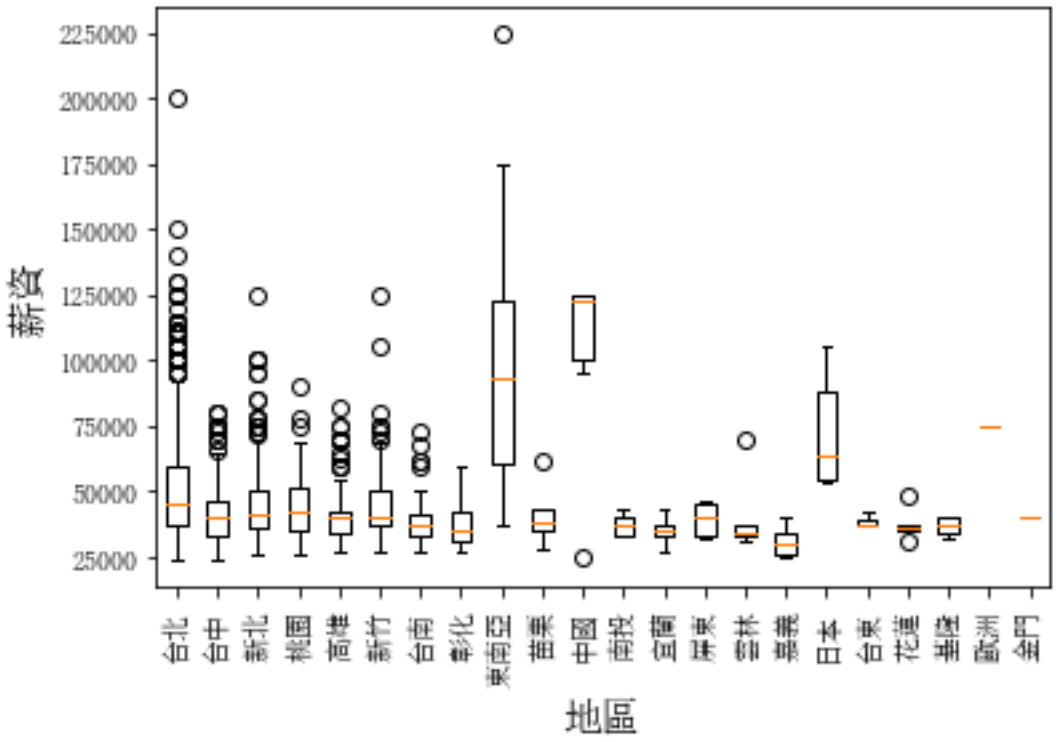

圖 4-5 地區與薪資箱型圖 (圖中的橘線為中位數)

三、管理責任

若該職缺需負擔管理責任，則不論管理人數為何，均視為同一類別，將其轉換為二元變數，由圖 4-6 可見需負擔管理責任的案件非常稀少，可能的原因有很多，例如，相較於不用負擔管理責任的工作，公司較不會傾向對外公開徵求需負擔管理責任的工作職缺。

DOI:10.6814/NCCU202001406

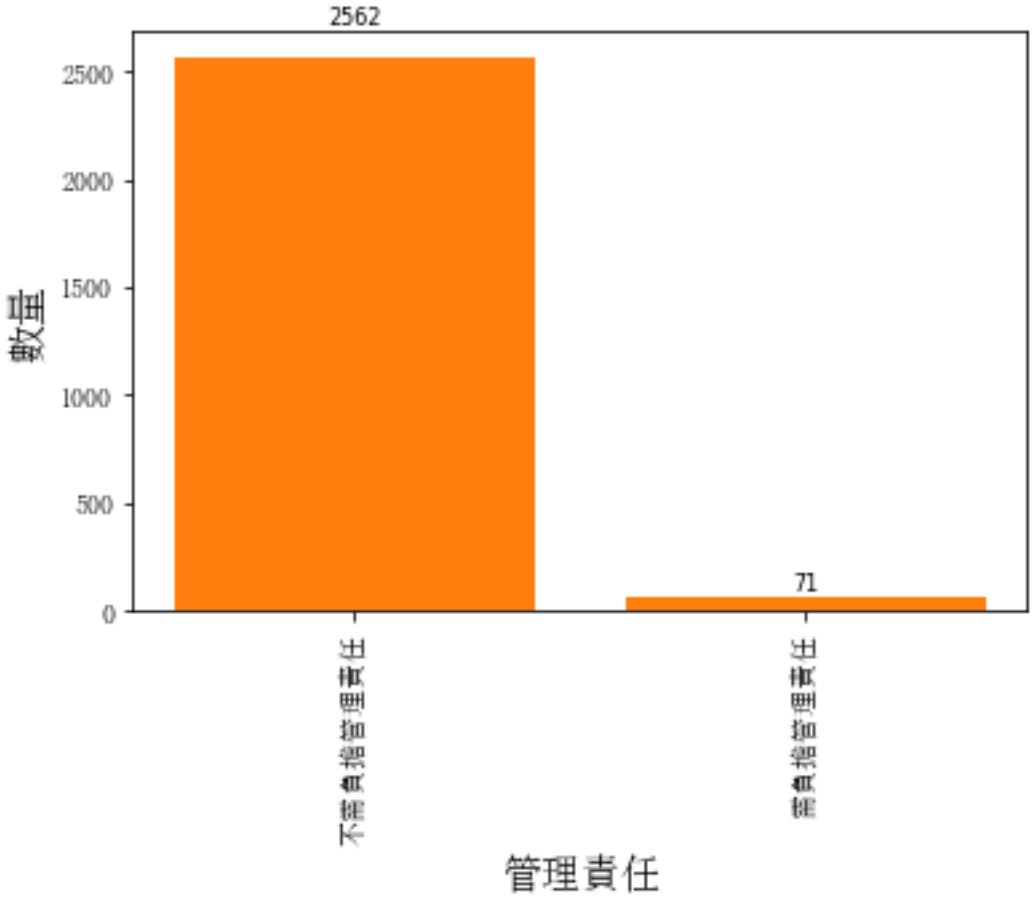

圖 4-6 管理責任分布柱狀圖

由圖 4-7 可以發現需負擔管理責任的工作案件分布稍微右偏，且根據 Mann-Whitney rank test 的檢定結果，需負擔管理責任的工作案件在顯著水準為 0.01 的情況下，薪資中位數顯著大於不需負擔管理責任的工作案件。而不需負擔管理責任的工作案件也稍微右偏，並存在許多離群值，表示雖然不需負擔管理責任的相關職缺雖然普遍薪資偏低，但仍存有高薪機會。

DOI:10.6814/NCCU202001406

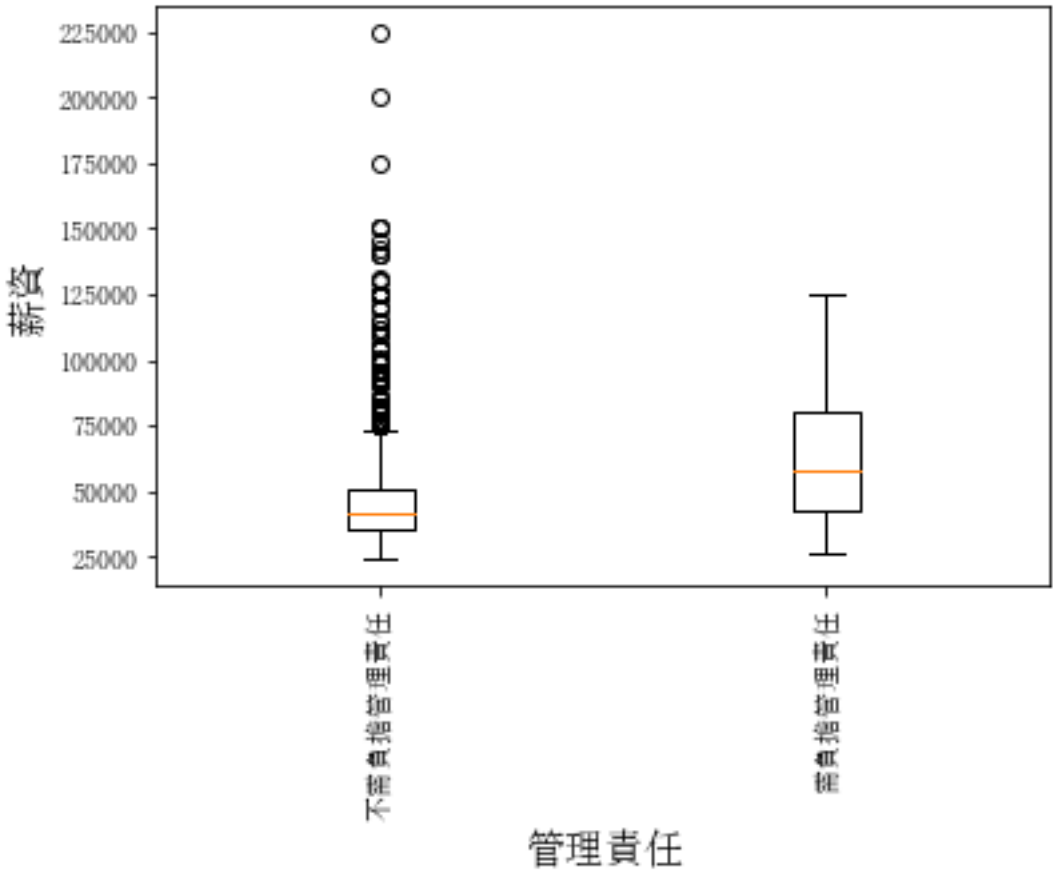

圖 4-7 管理責任與薪資箱型圖 (圖中的橘線為中位數)

四、出差要求

若該職缺需出差，根據時間長短，分為 6 個類別，從圖 4-8 中可以發現大多數的工作案件均無出差的要求。

DOI:10.6814/NCCU202001406

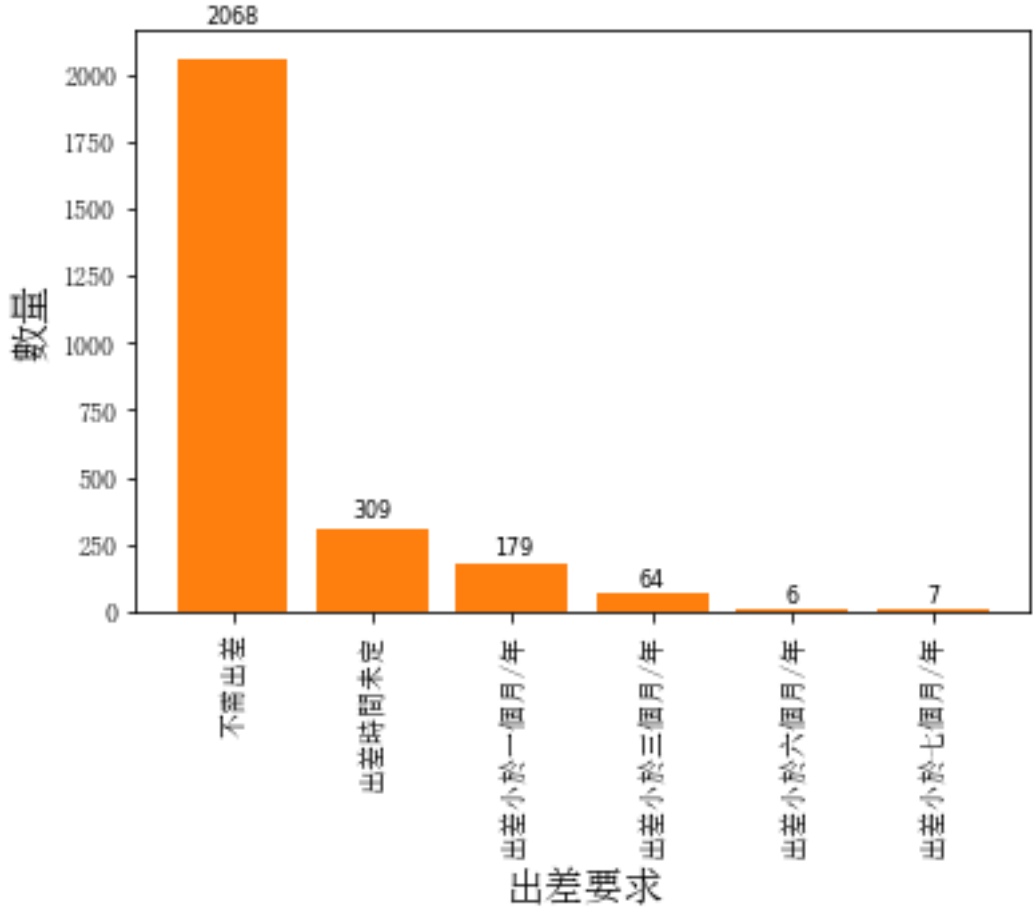

圖 4-8 出差要求分布柱狀圖

由圖 4-9 可以發現出差要求的多數類別存在高薪離群值，且根據 Mann-Whitney rank test 的檢定結果(見附錄表三)，在顯著水準為 0.01 的情況下，任意兩個出差要求時間的薪資中位數與其他類別的薪資中位數均不顯著有差異，這也反映了出差要求並不是高薪職缺的必要因素。

DOI:10.6814/NCCU202001406

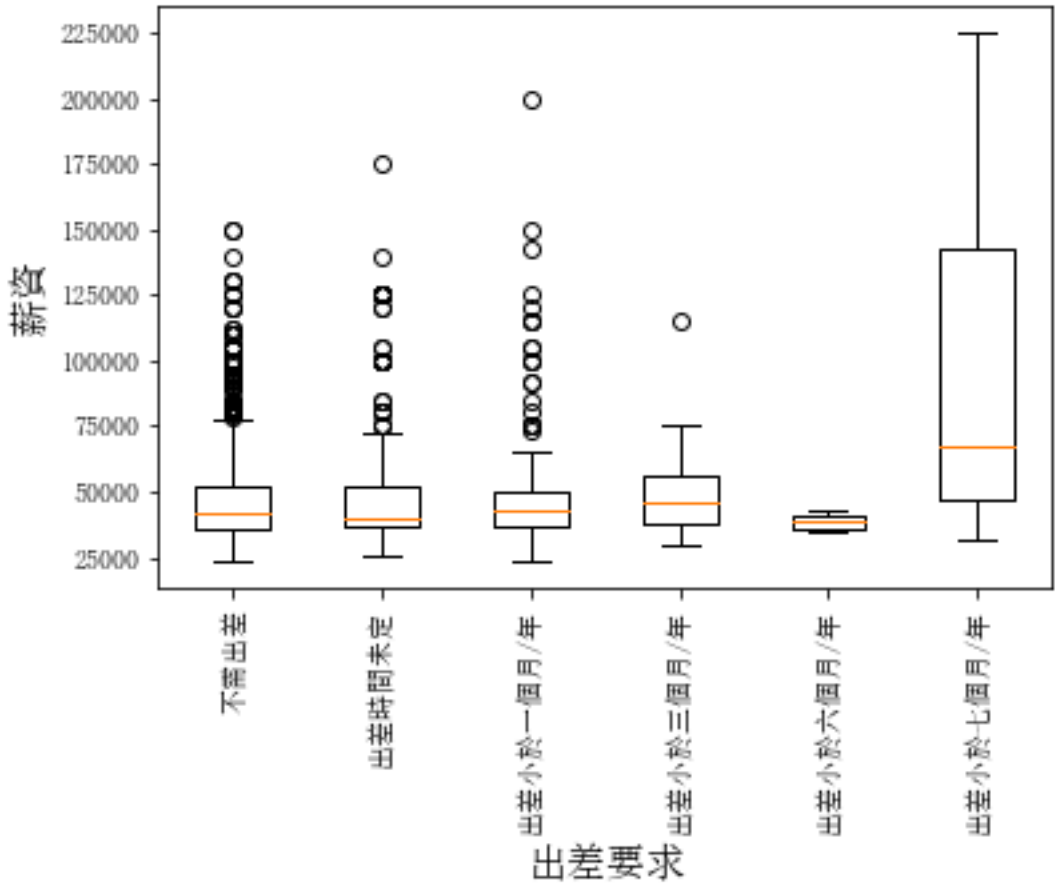

圖 4-9 出差要求與薪資箱型圖 (圖中的橘線為中位數)

五、上班時間

若上班時間未提及需輪班，則視為不需輪班，不論其時間區段分布、長度，均整合為同一類別，使其轉換為二元變數，如圖 4-10 所示，不需輪班的案件數量遠多於需輪班。

DOI:10.6814/NCCU202001406

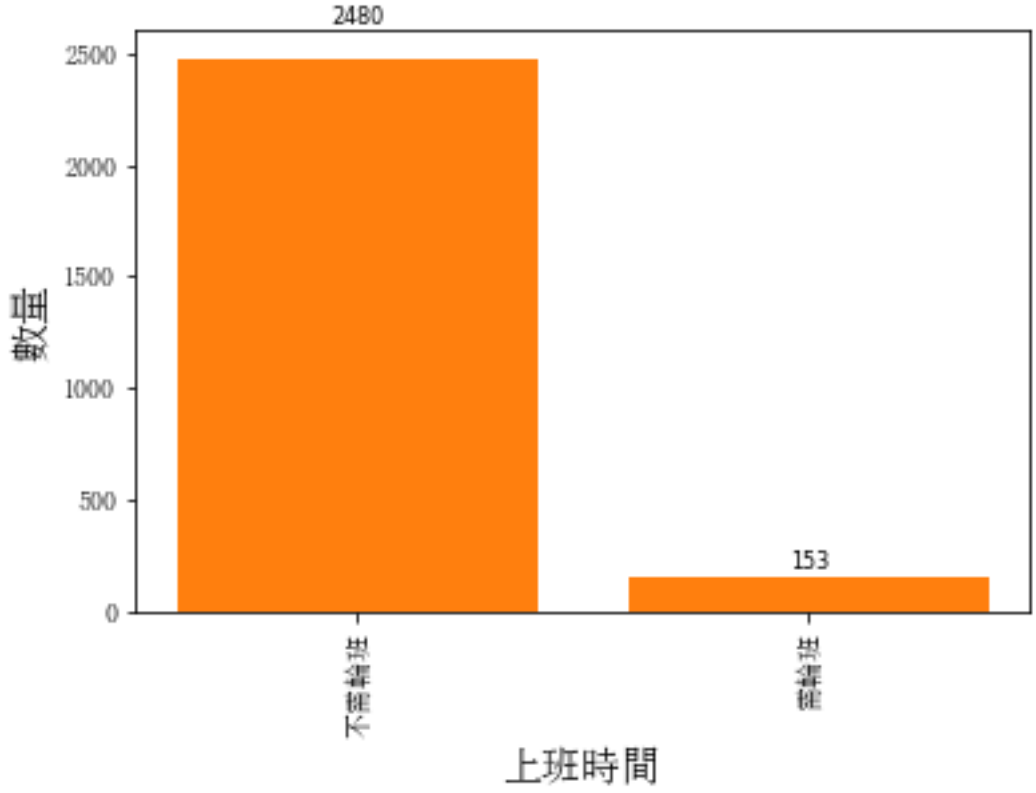

圖 4-10 上班時間分布柱狀圖

由圖 4-11 可以發現不需輪班較需輪班的分布較為右偏，且有較多的高薪離群值的工作案件，且根據 Mann-Whitney rank test 的檢定結果，在顯著水準為 0.01 的情況下，兩種類別的薪資中位數不顯著有差異。

DOI:10.6814/NCCU202001406

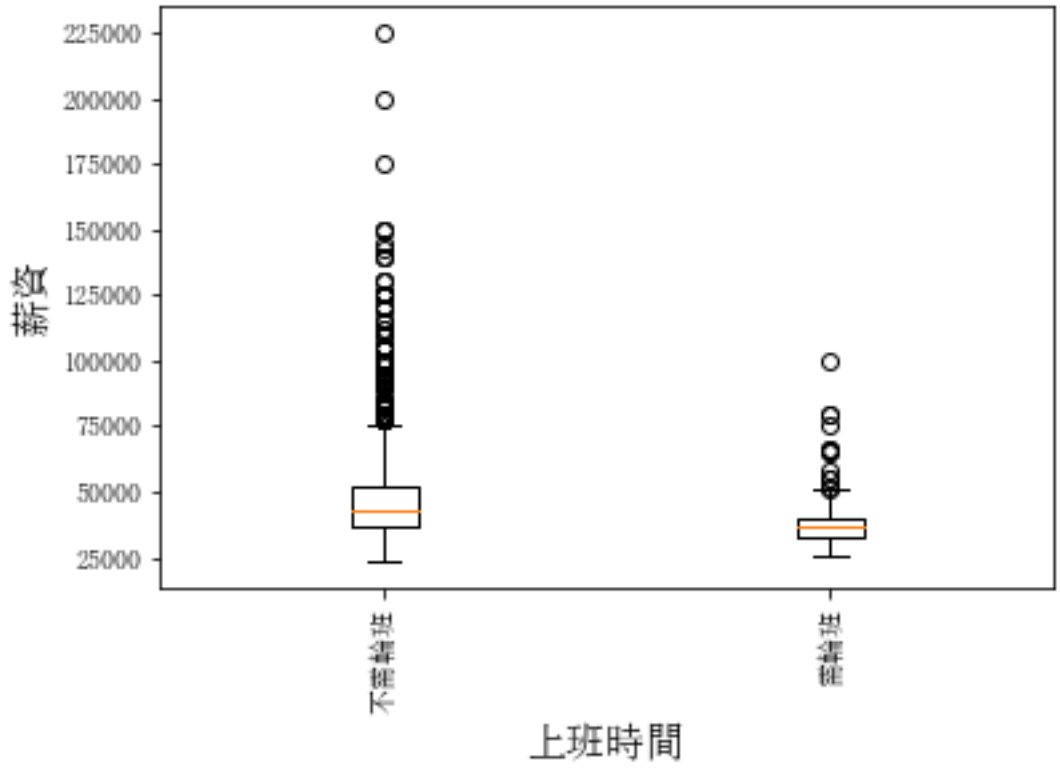

圖 4-11 上班時間與薪資箱型圖 (圖中的橘線為中位數)

六、周休

該變數在資料集中本身即為"周休二日"與"依公司規定"的二元變數，其數量分布如圖 4-12 所示。

DOI:10.6814/NCCU202001406

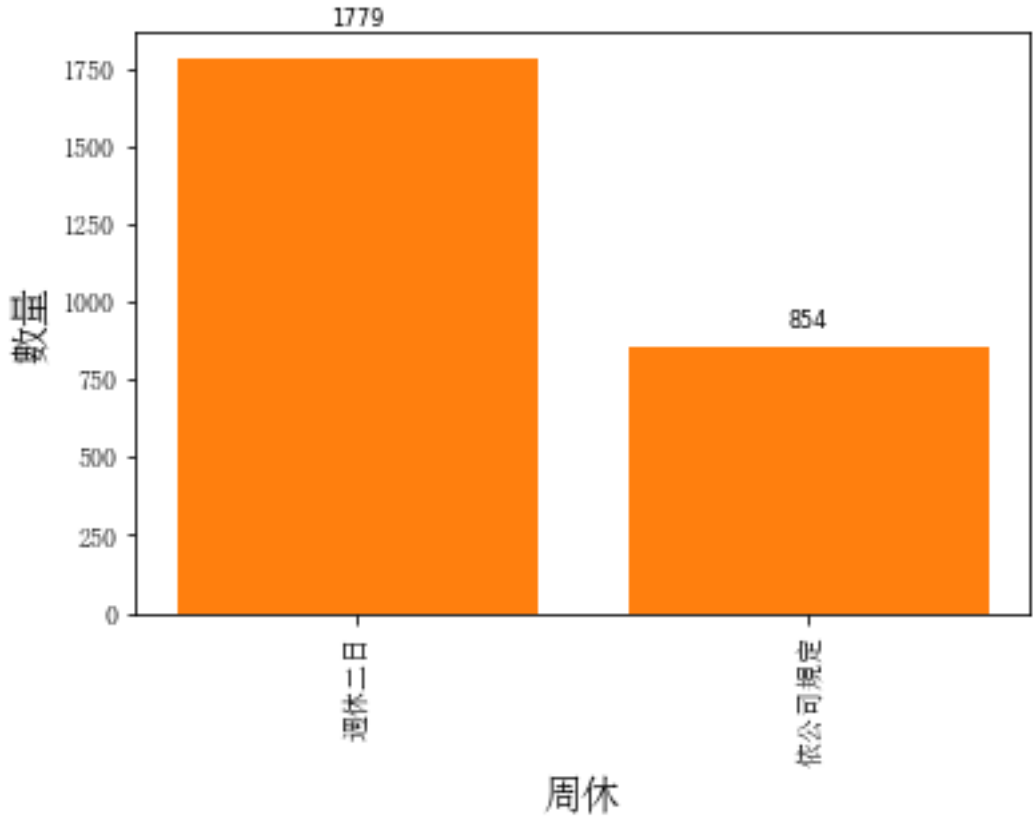

圖 4-12 周休分布柱狀圖

由圖 4-13 可以發現周休的兩種類別均為右偏，並有許多的高薪離群值，且根據 Mann-Whitney rank test 的檢定結果，在顯著水準為 0.01 的情況下，兩種類別的薪資中位數不顯著有差異，且與原始的薪資分布一樣。

DOI:10.6814/NCCU202001406

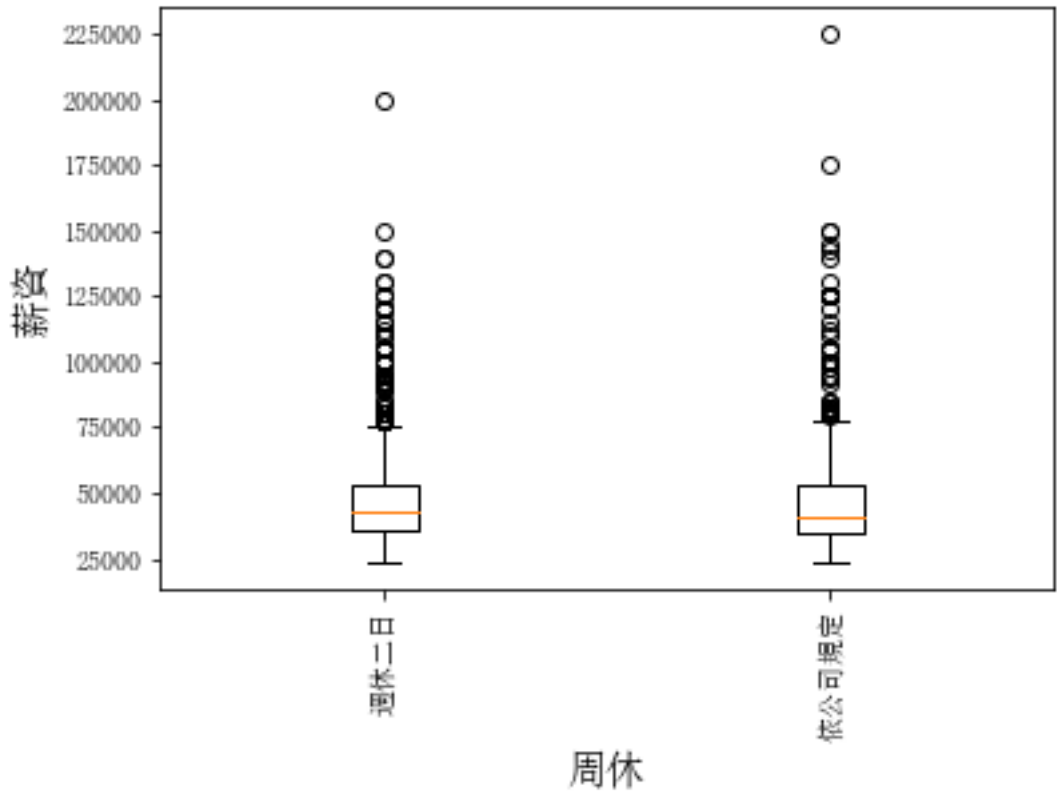

圖 4-13 周休與薪資箱型圖 (圖中的橘線為中位數)

七、經歷

該變數以經歷年資差異做為區隔，為一類別變數，如圖 4-14 可見工作案件數量大略隨著經歷年資的增加而遞減。

DOI:10.6814/NCCU202001406

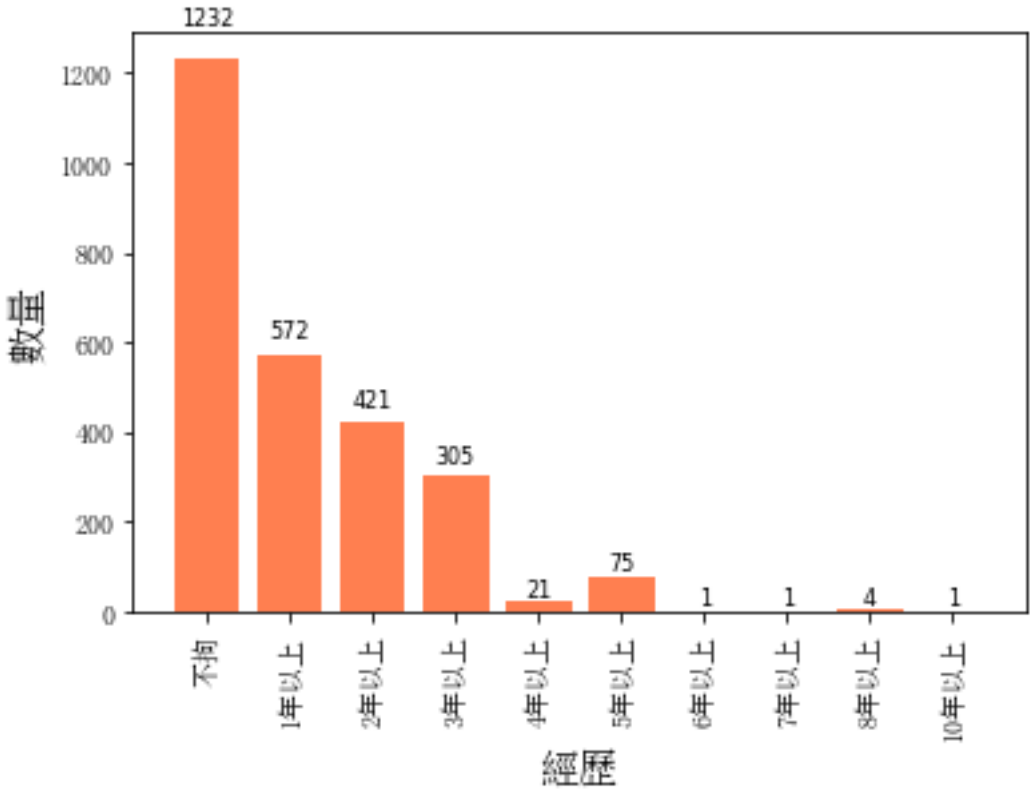

圖 4-14 經歷分布柱狀圖

由圖 4-15 中可見，薪資水準大略有隨著經歷年資增加而有增加的趨勢，但在不要求經歷的案件中，仍有高薪水準離群值的工作案件。但根據 Mann-Whitney rank test 的檢定結果(見附錄表四)，在顯著水準為 0.01 的情況下，部份類別例如經歷在四年以下的薪資中位數各自是顯著有差異的，但在經歷超過四年以上之後的薪資中位數則不顯著有差異。這樣的結果與劉姿君(1993)和周宜滿(2004)的研究結果相似，也就是在台灣地區工作經驗的確對於薪資有顯著的影響，而這樣的影響在資訊軟體系統類的相關職缺也同樣適用。另外，Mart´ın et al.(2018)使用西班牙的資料集，檢視資訊軟體系統類的相關職缺，同樣也可以發現工作經驗對於薪資的顯著影響。

DOI:10.6814/NCCU202001406

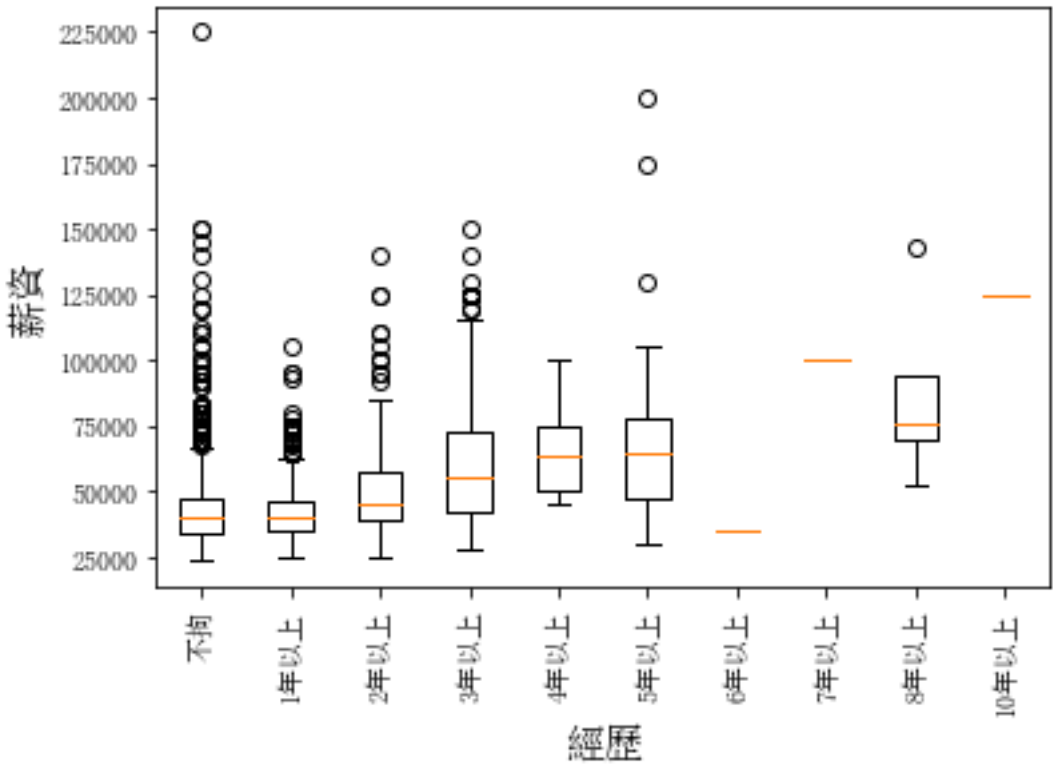

圖 4-15 經歷與薪資箱型圖 (圖中的橘線為中位數)

八、學歷

因為同一徵才案件中可能會並列不同的學位要求，因此在此以高中→專科→大學→碩士→博士的順序作篩選，若位於前面的學位有出現，則將該則案例的學歷要求歸於該項。例如高中、大學若同時出現在學歷要求中，則該案件會歸類於高中。由圖 4-16 可見工作案件主要以專科與大學為主，研究所以上的學歷要求則相對稀少。

DOI:10.6814/NCCU202001406

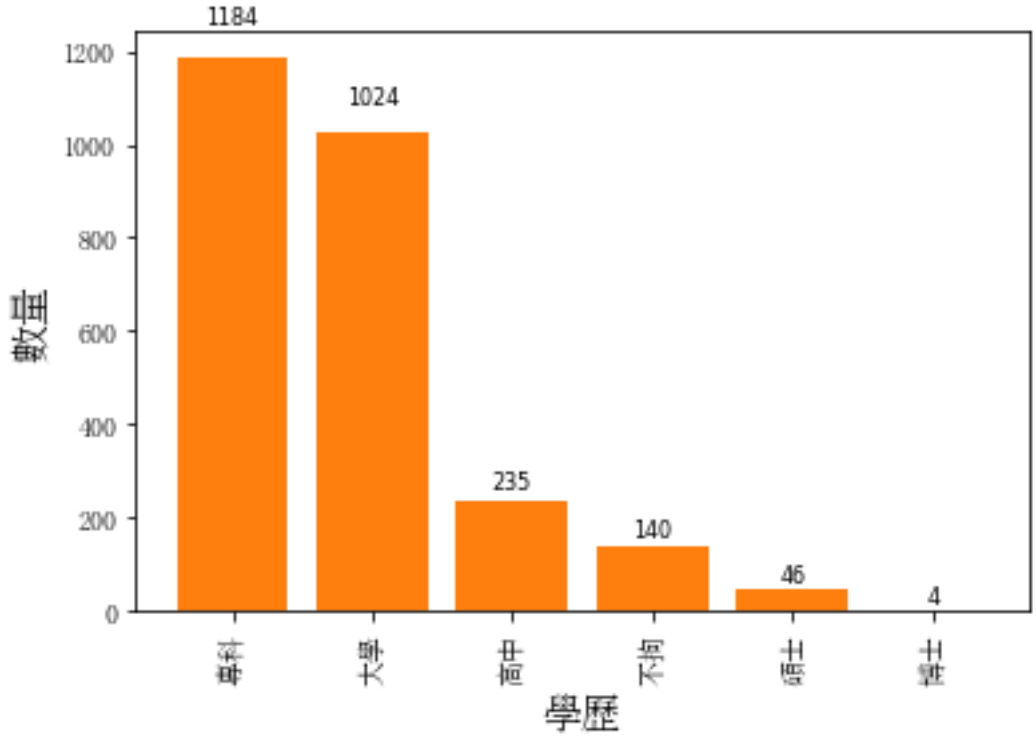

圖 4-16 學歷分布柱狀圖

根據 Mann-Whitney rank test 的檢定結果(見附錄表五)，在顯著水準為 0.01 的情況下，需要高中和大學學歷的工作案件之間的薪資中位數差異並不大，但是，需要高中學歷的工作案件與大學及碩士的工作案件的薪資中位數差異很大，專科和大學與碩士的工作案件之間的薪資中位數差異也是如此。這樣的結果也實屬合理，因為資訊軟體系統類的相關職缺所涉及的技術，通常是在大學所教授，因此大學的學歷對於企業而言是一種不可或缺的求職者能力的保證，進而造成如此薪資上的差異。這樣的結果與劉姿君(1993)和周宜滿(2004)的研究結果相似，即台灣地區的教育程度對於薪資具有顯著的影響，而在此研究中也發現對於資訊軟體系統類的工作職缺也是如此；但是 Mart´ın et al.(2018)使用西班牙的資訊軟體系統類的職缺資料集，則發現教育年數對於薪資水準並不特別顯著。此外，也可以從圖 4-17 中發現專科與大學都有相當多的高薪離群值。

DOI:10.6814/NCCU202001406

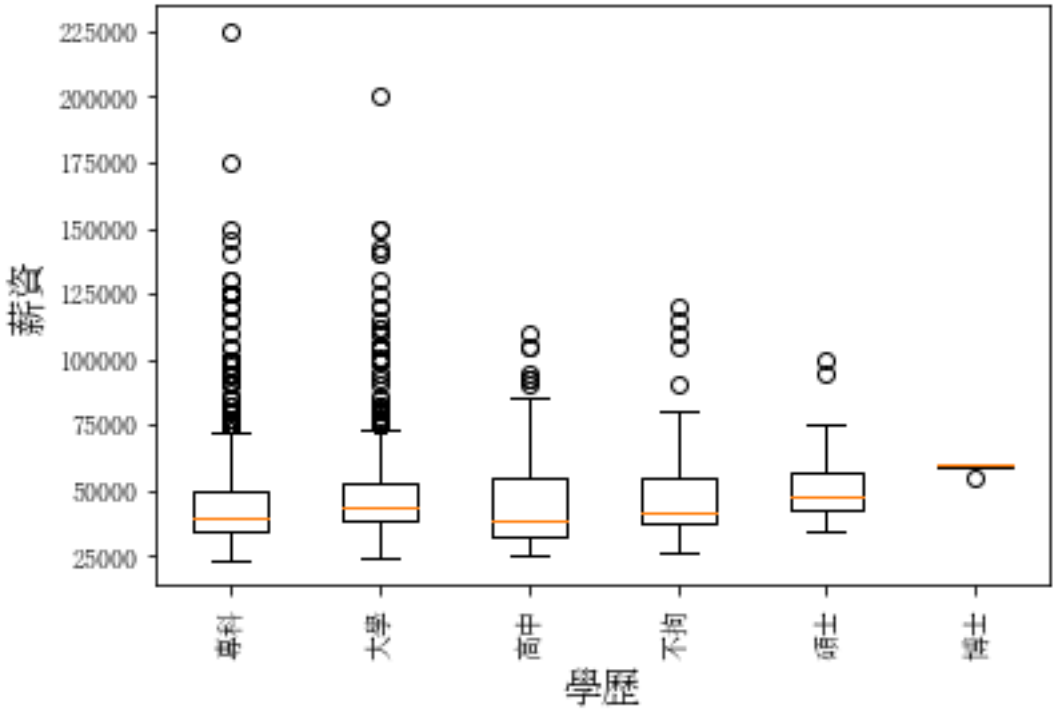

圖 4-17 學歷與薪資箱型圖 (圖中的橘線為中位數)

九、科系

將科系的欄位以 one-hot encoding 作轉換，變成 68 個相異欄位，許多科系的數量落在個位數。從圖 4-18 到圖 4-21 中可以發現數量以資訊類科系較多，例如資訊工程、資訊管理。

DOI:10.6814/NCCU202001406

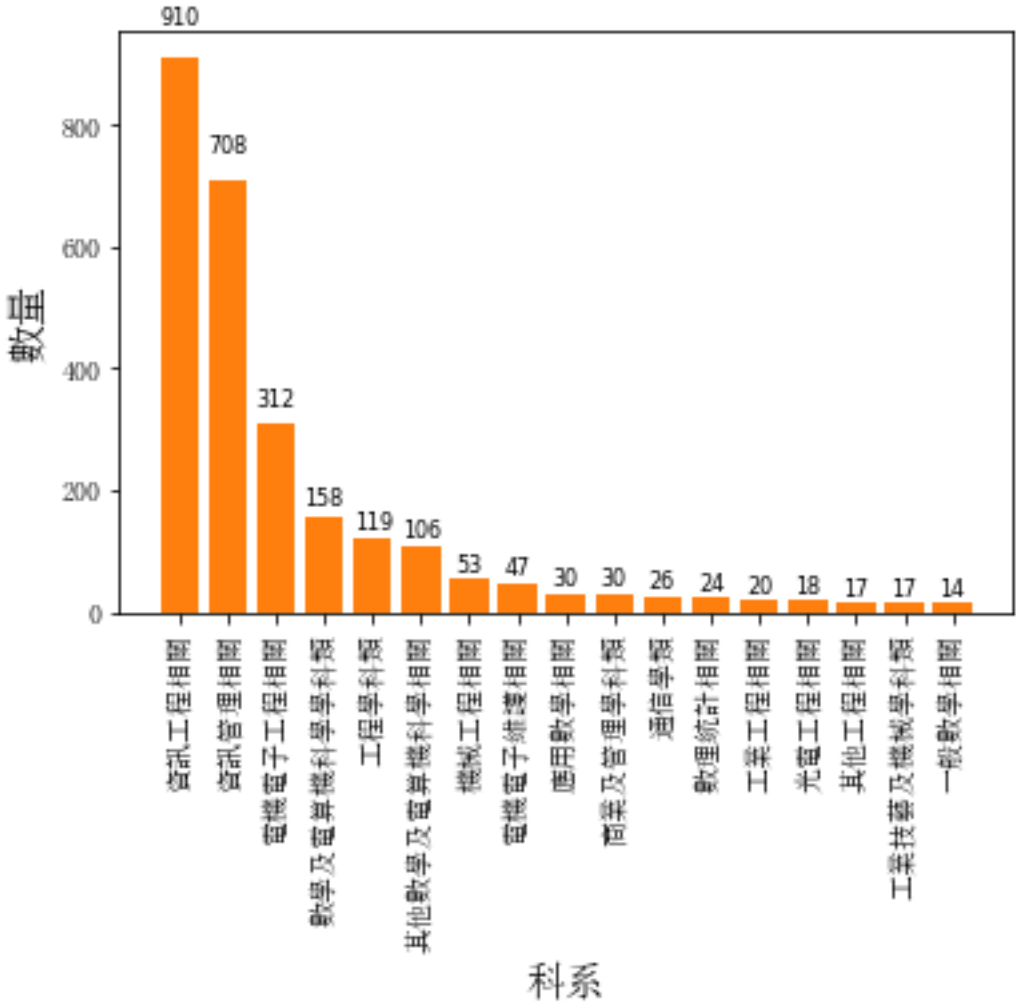

圖 4-18 科系分布柱狀圖

DOI:10.6814/NCCU202001406

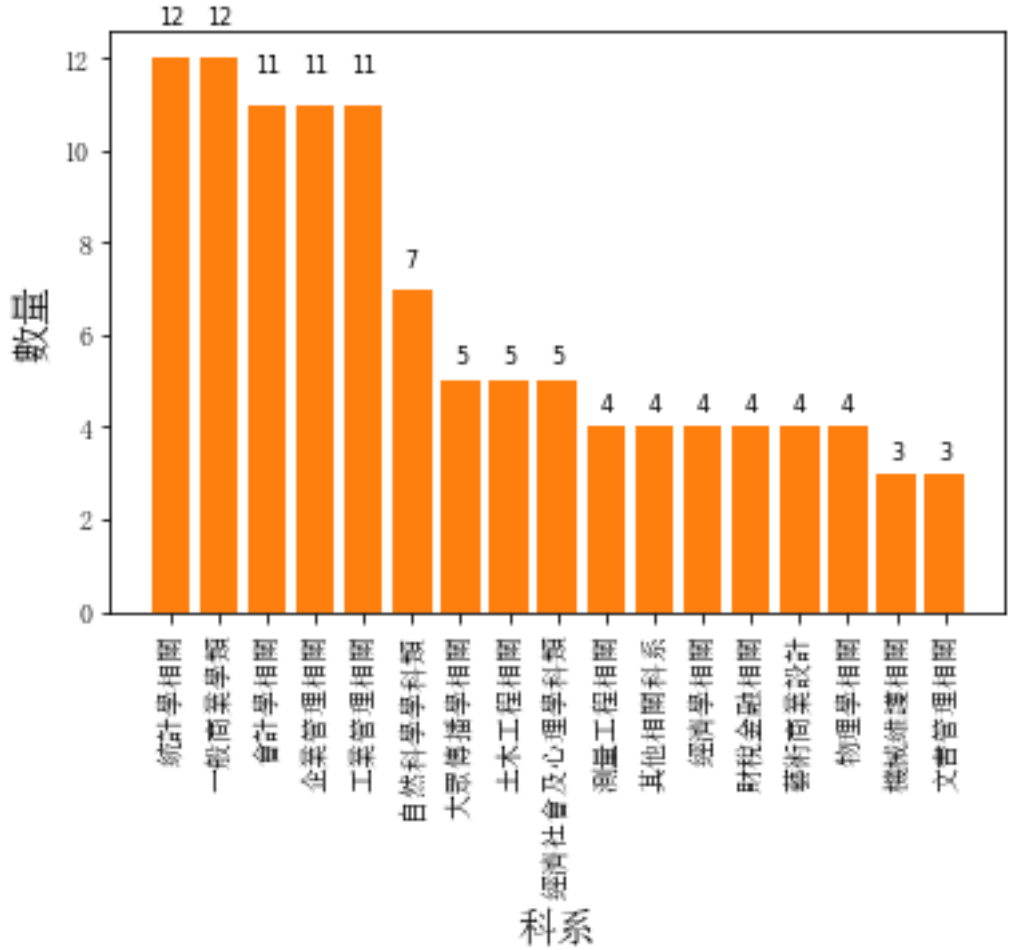

圖 4-19 科系分布柱狀圖(續)

DOI:10.6814/NCCU202001406

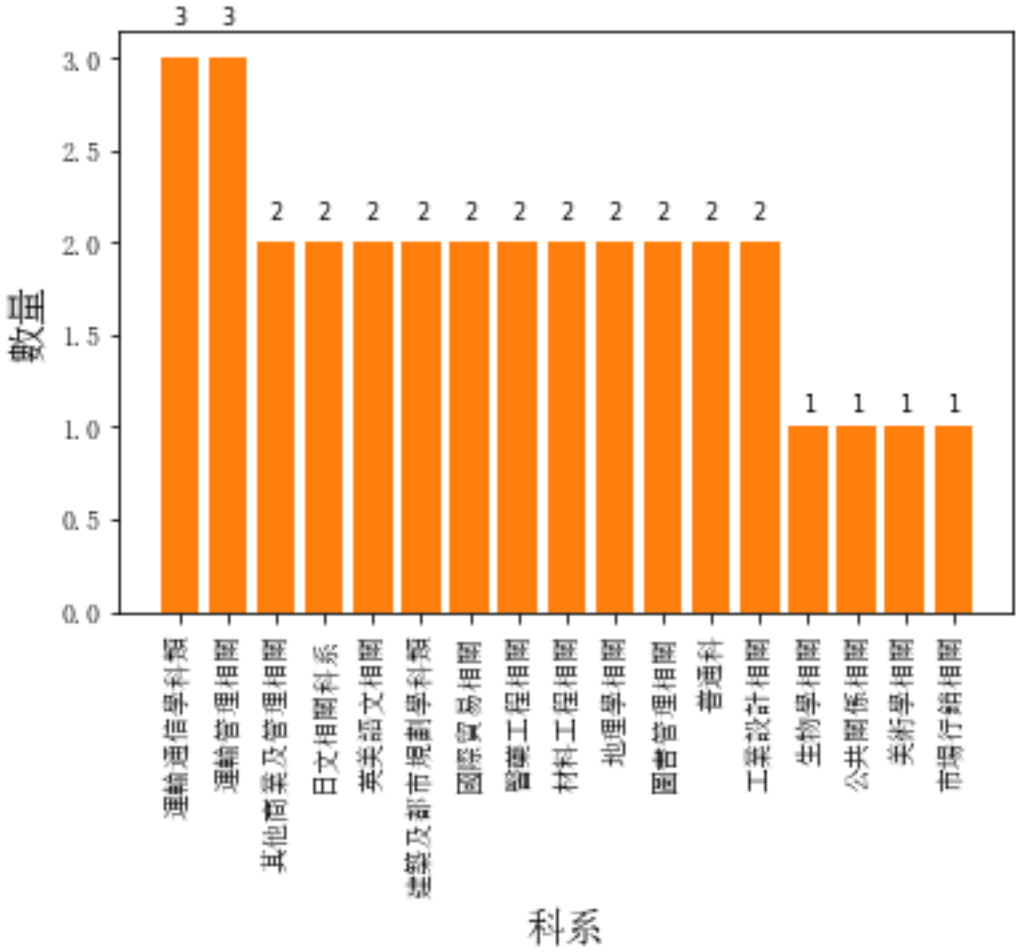

圖 4-20 科系分布柱狀圖(續)

DOI:10.6814/NCCU202001406

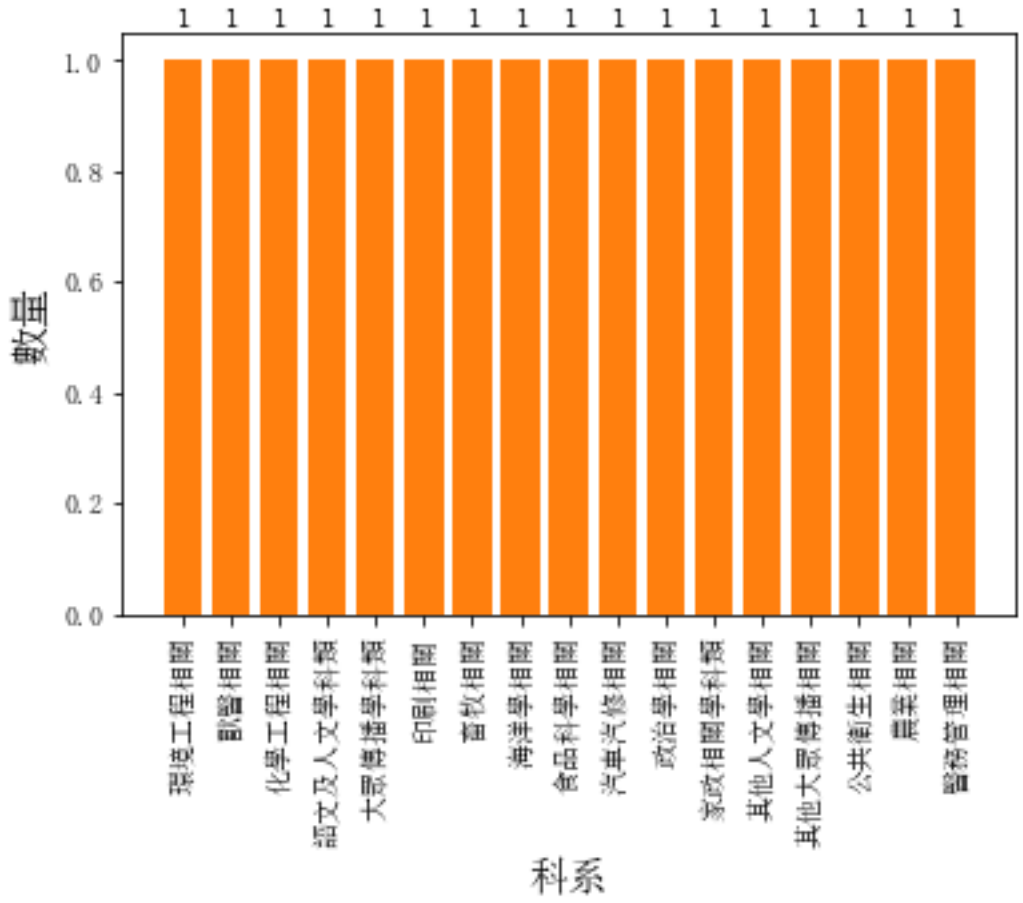

圖 4-21 科系分布柱狀圖(續)

從圖 4-22 到圖 4-25 中可以發現，各科系的薪資分布多為右偏分布，且薪資中位數稍有差異，其中以數學及工程相關科系略高。此外，資訊相關科系雖然薪資中位數普遍較低，但仍有許多高薪離群值。例如根據 Mann-Whitney rank test 的檢定結果，在顯著水準為 0.01 的情況下，數學及電算機科學的薪資中位數顯著高於資訊工程相關學系。這樣的結果與林鼎晃(2013)的研究結果相似，即在台灣地區科系差異對於薪資水準有顯著影響，本研究證實了針對台灣地區的資訊軟體系統類職缺，科系差異對於薪資水準同樣也有顯著影響。

DOI:10.6814/NCCU202001406

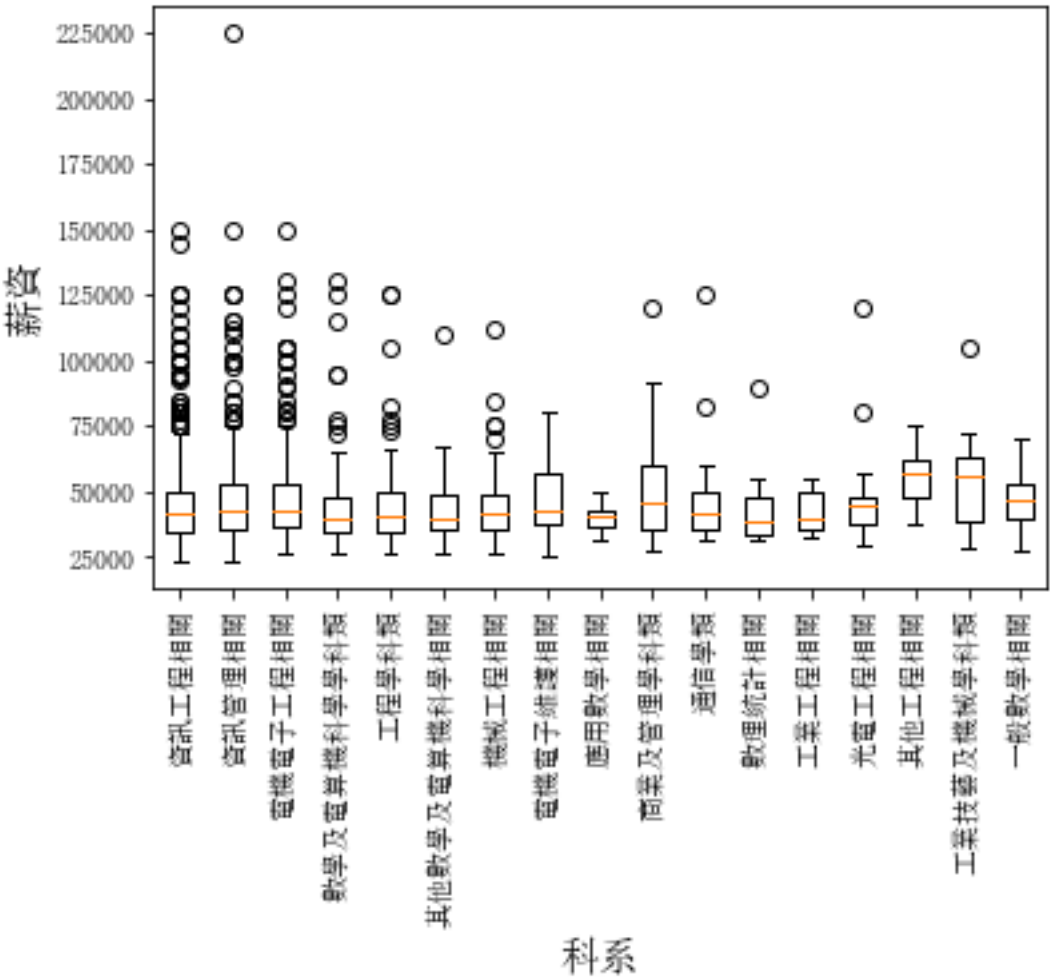

圖 4-22 科系與薪資箱型圖 (圖中的橘線為中位數)

DOI:10.6814/NCCU202001406

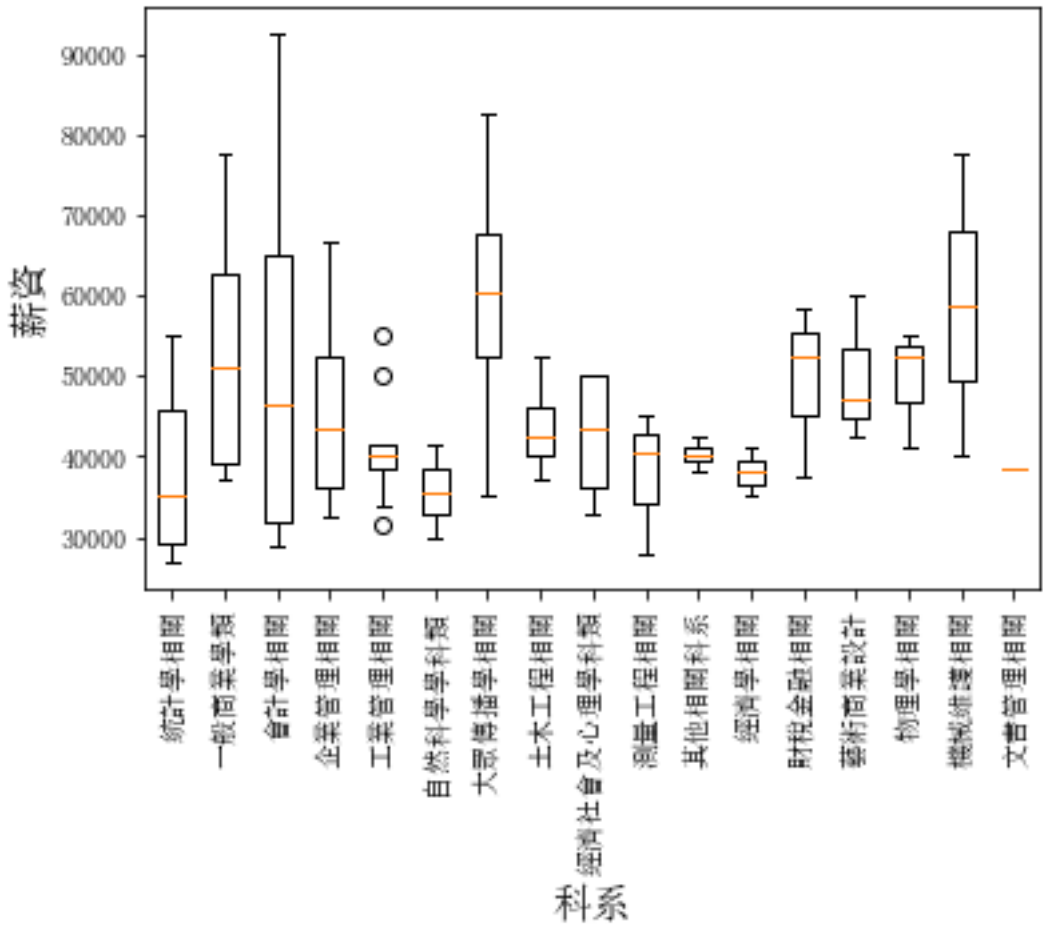

圖 4-23 科系與薪資箱型圖(續) (圖中的橘線為中位數)

DOI:10.6814/NCCU202001406

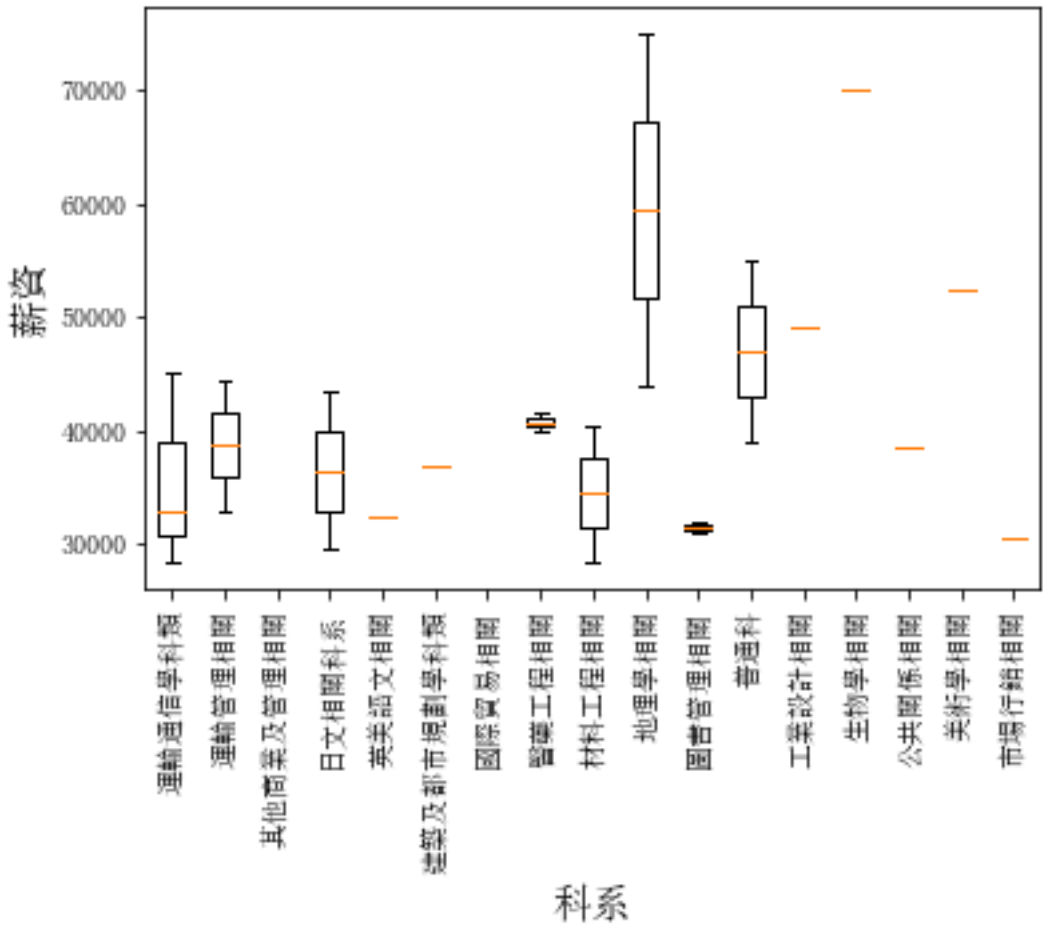

圖 4-24 科系與薪資箱型圖(續) (圖中的橘線為中位數)

DOI:10.6814/NCCU202001406

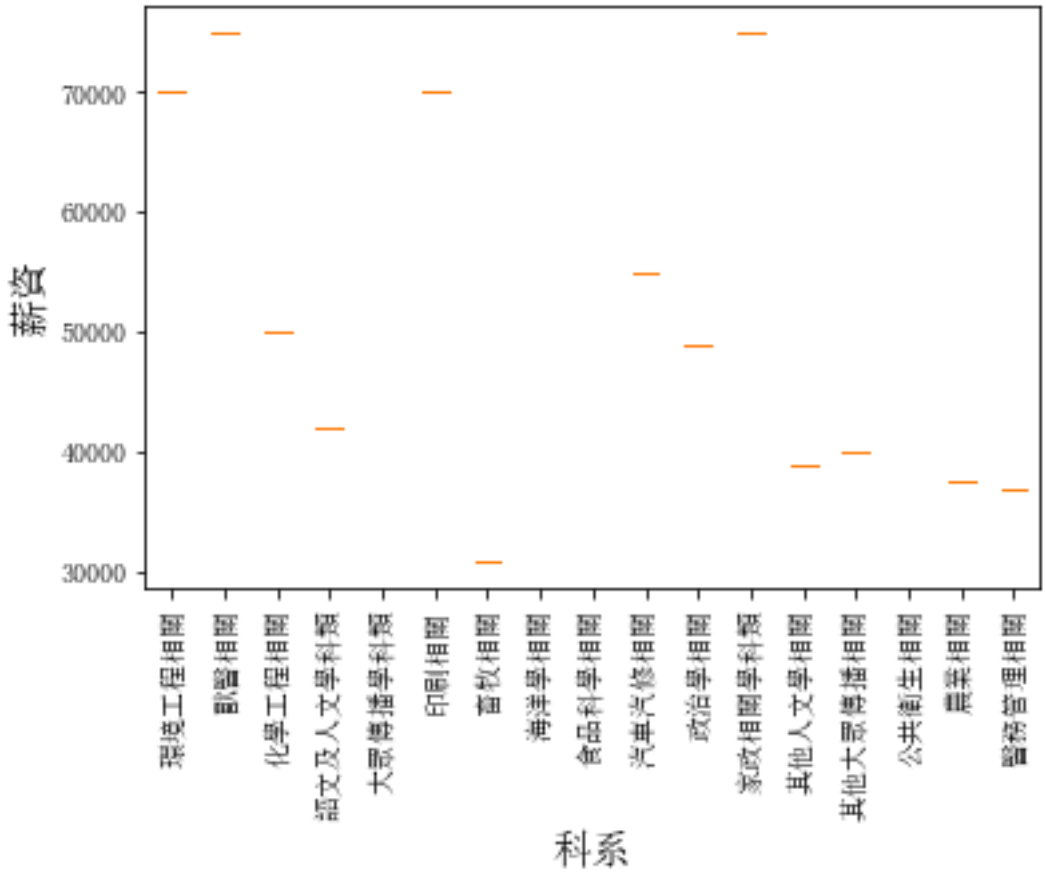

圖 4-25 科系與薪資箱型圖(續) (圖中的橘線為中位數)

十、語言

在語言方面以英文與不拘作為關鍵字篩選，剩餘的其他案件均歸類為其他語言如日文、韓文等等，整理後剩餘三個欄位，由圖 4-26 中可見要求英文的工作案件幾乎與不要求語言能力的工作案件數量相同，此外，其他語言的要求相對於前兩者而言則非常罕見。

DOI:10.6814/NCCU202001406

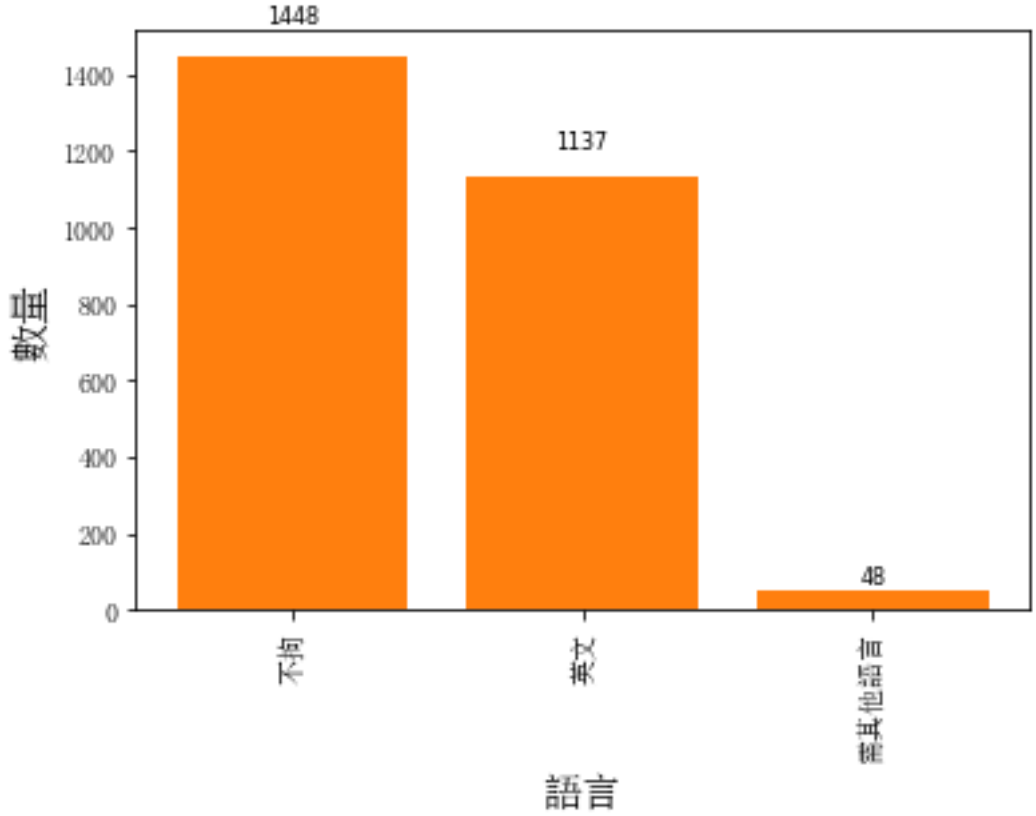

圖 4-26 語言分布柱狀圖

由圖 4-27 可以發現不拘與英文的類別與原薪資分布相似，均有高薪資水準的離群值。且根據 Mann-Whitney rank test 的檢定結果，在顯著水準為 0.01 的情況下，需其他語言的薪資中位數顯著低於其他兩個類別，同時，不拘語言與要求英文能力的薪資中位數沒有顯著差異，因此可以推論資訊軟體系統的相關工作，語言能力並不是決定薪資水準的重要因素。這樣的結果與 Singh(2016)針對於印度的資訊軟體系統類職缺的研究結果有所差異，在 Singh(2016)的研究中，英文能力對於薪資是有顯著的影響的，在本研究則看不到相似的研究結果。

DOI:10.6814/NCCU202001406

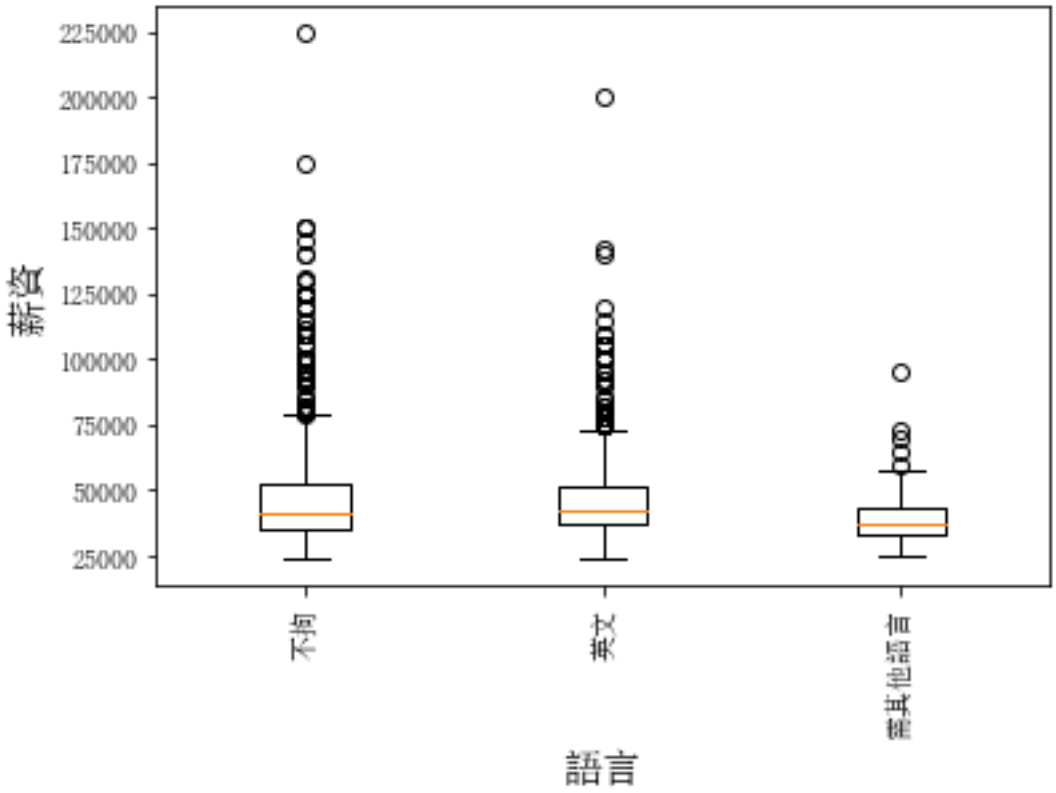

圖 4-27 語言與薪資箱型圖 (圖中的橘線為中位數)

十一、擅長工具

將擅長工具的欄位以 one-hot encoding 作轉換，變成 299 個相異之工具欄位，因為種類過於多樣，故在圖 4-28 中將數量較少的擅長工具整併為其他。由圖中可以發現台灣仍以涉及網頁的程式語言為主流例如 JavaScript、HTML，且由其他類別的數量可知，資訊軟體系統類的工作案件的使用工具非常多樣化。

DOI:10.6814/NCCU202001406

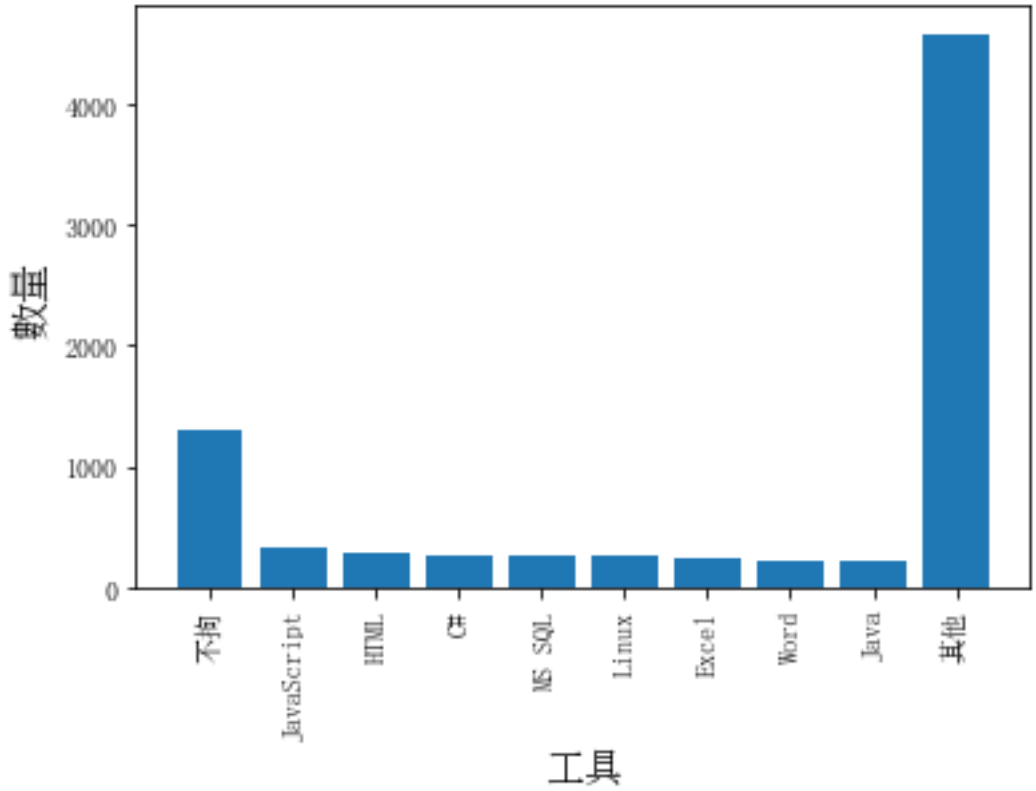

圖 4-28 擅長工具分布柱狀圖 (註: 擷取出現較多的擅長工具，其餘則併入其他欄位)

由圖 4-29 可以發現，大部分工具要求的薪資中位數無太大差異，且多呈現右偏分布，並有很多的高薪離群值。例如根據 Mann-Whitney rank test 的檢定結果，在顯著水準為 0.01 的情況下，JavaScript 的薪資中位數不顯著異於 HTML。而同樣在顯著水準為 0.01 的情況下，C#則顯著高於 HTML。在第五章，則會透過建構薪資預測模型，分析擅長工作的差異對於薪資的影響。

DOI:10.6814/NCCU202001406

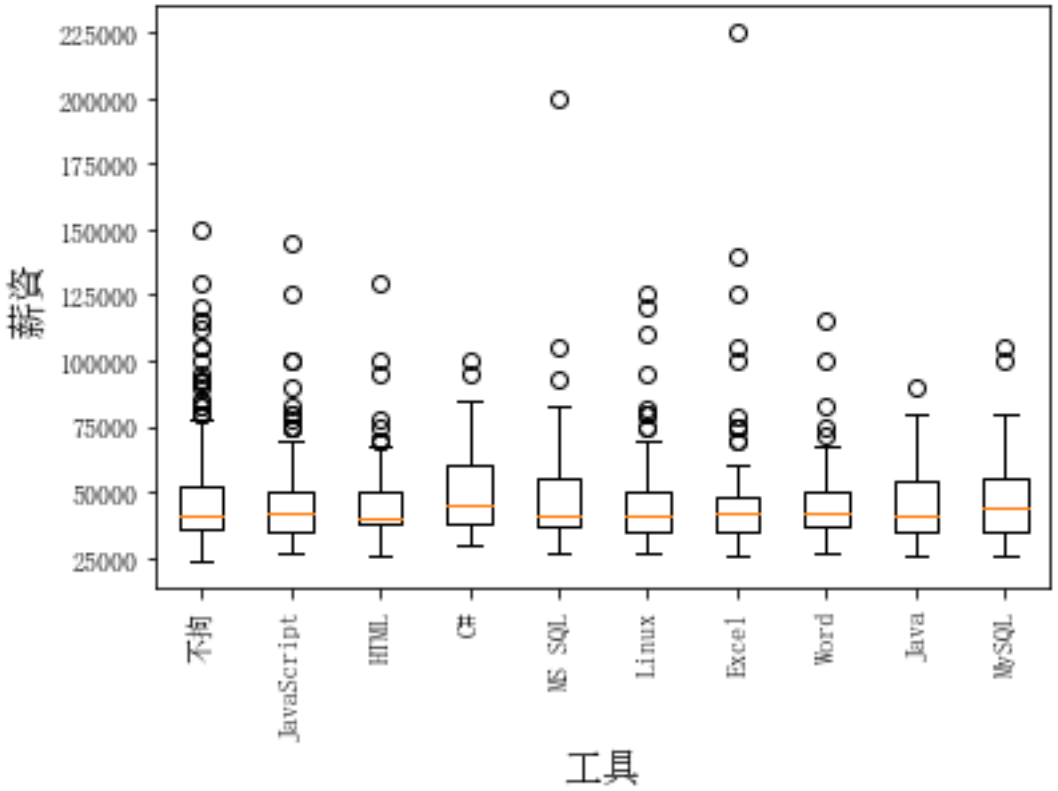

圖 4-29 擅長工具與薪資箱型圖 (圖中的橘線為中位數) (註: 圖中擷取在資料集中出現較多的擅長工具，由左至右出現的頻率為遞減)

十二、工作技能

將工作技能以 one-hot encoding 作轉換，變成 184 個相異之工作技能欄位，由圖 4-30 可以發現資訊軟體系統類的工作技能非常多樣化。

DOI:10.6814/NCCU202001406

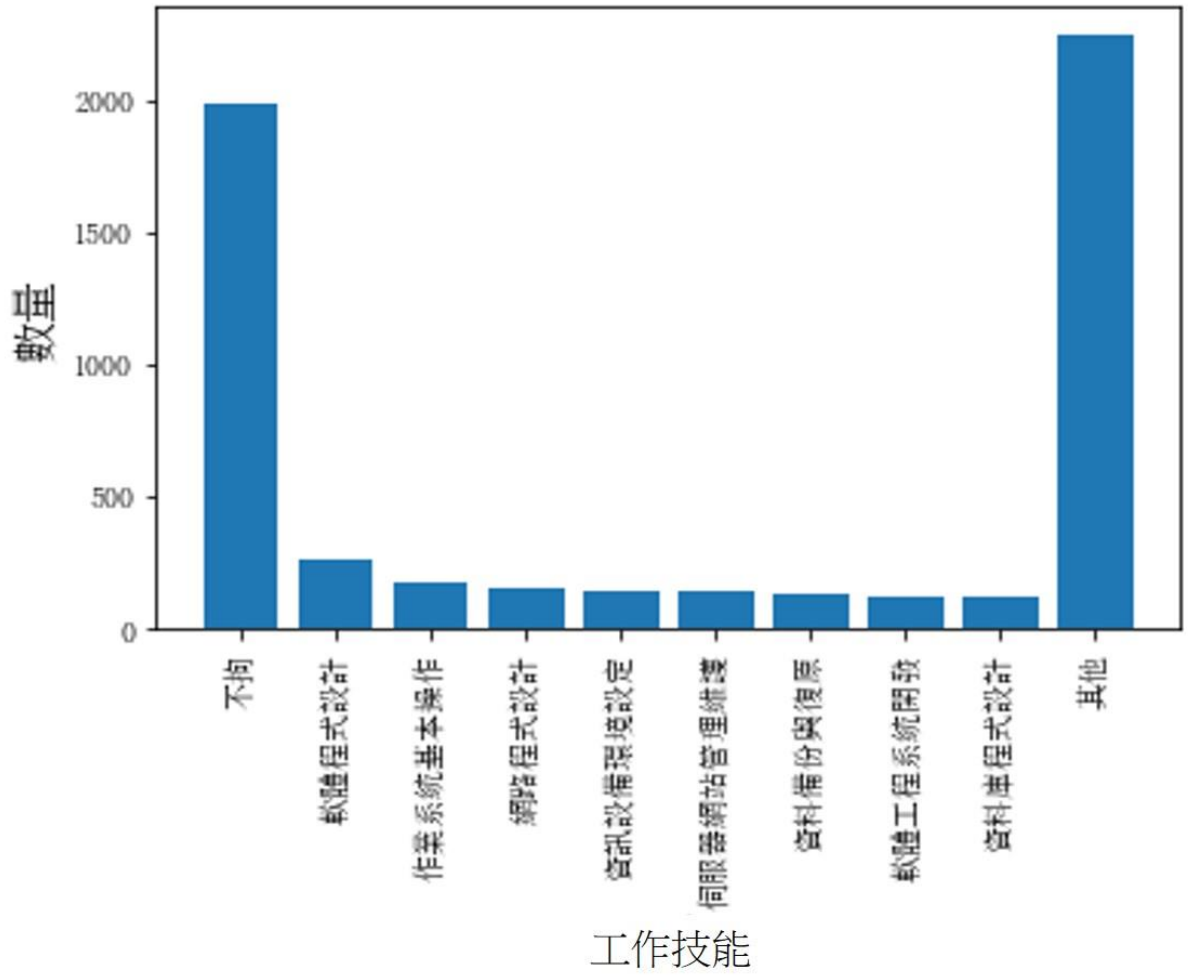

圖 4-30 工作技能分布柱狀圖(註: 擷取出現較多的工作技能，其餘則併入其他欄位)

由圖 4-31 可以發現，大部分的工作技能要求的薪資中位數之間有些許差異，且大多呈現右偏分布，並有很多高薪離群值。例如根據 Mann-Whitney rank test 的檢定結果，在顯著水準為 0.01 的情況下，作業系統基本操作的薪資中位數顯著高於軟體程式設計。相反地，同樣在顯著水準為 0.01 的情況下，資訊設備環境設定的中位數與伺服器網站管理維護的中位數是沒有顯著差異的。在第五章，則會透過建構薪資預測模型，分析工作技能的差異對於薪資的影響。

DOI:10.6814/NCCU202001406

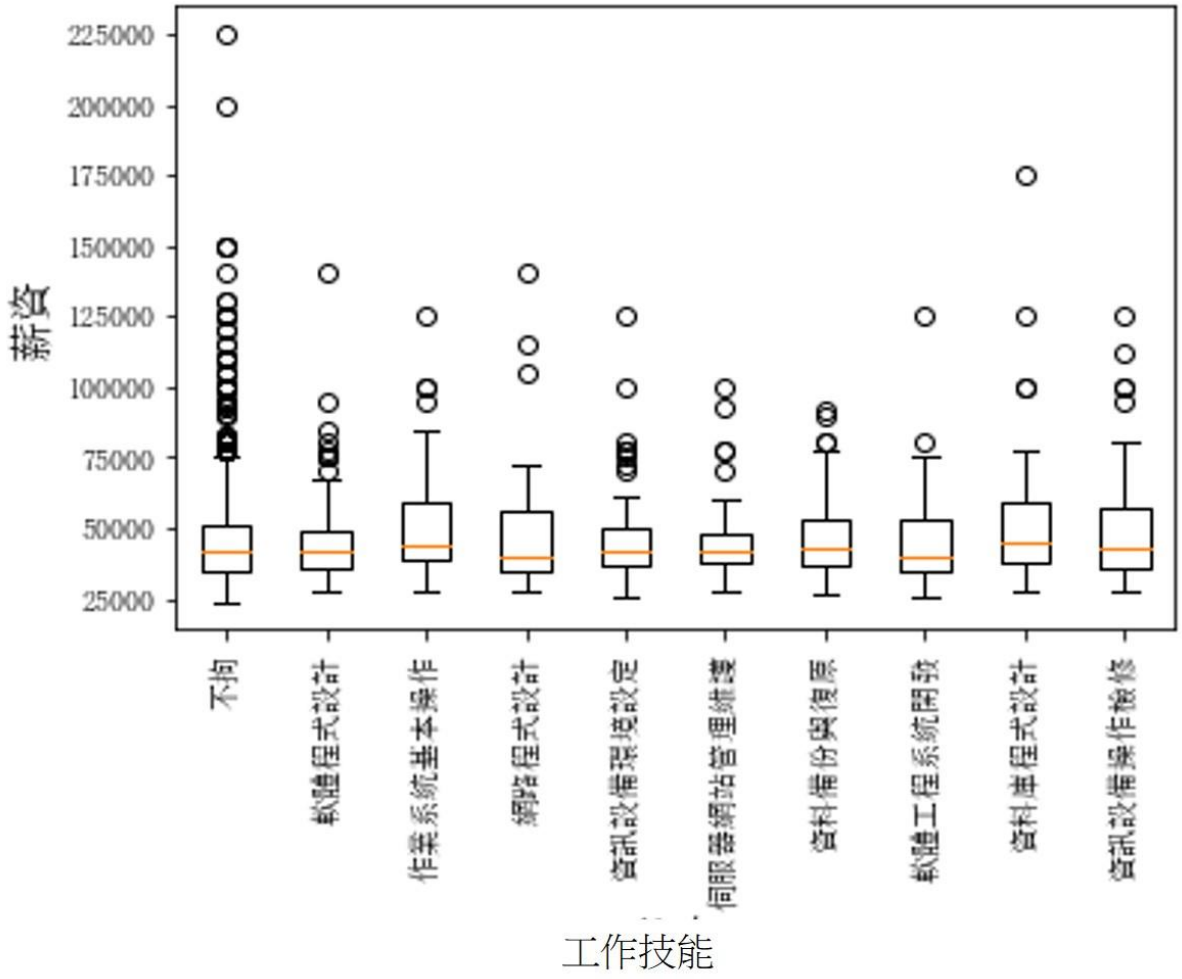

圖 4-31 工作技能與薪資箱型圖 (圖中的橘線為中位數) (註: 圖中擷取在資料集中出現較多的工作技能，由左至右出現的頻率為遞減)

十三、職位

將職位以 one-hot encoding 作轉換，變成 160 個相異之職位欄位，由圖 4-32 可以發現資訊軟體系統類工作職位以軟體設計工程師為主，但從其他欄位的數量也可以發現其職位非常多樣化。

DOI:10.6814/NCCU202001406

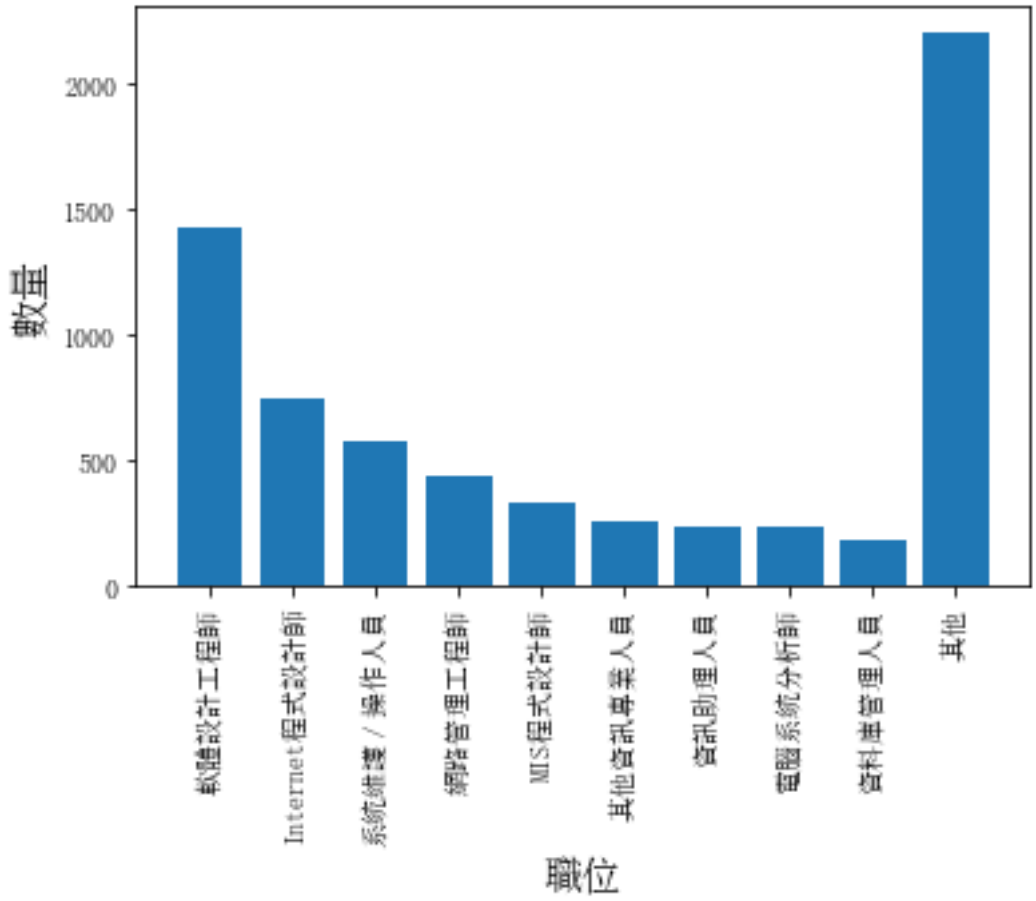

圖 4-32 職位分布柱狀圖(註: 擷取出現較多的職位，其餘則併入其他欄位)

由圖 4-33 可以發現，各種職位的薪資中位數大多無太大差異，且呈現右偏分布，並有很多高薪離群值。例如根據 Mann-Whitney rank test 的檢定結果，在顯著水準為 0.01 的情況下，軟體設計工程師的薪資中位數與 Internet 程式設計師無顯著差異。相反地，MIS 程式設計師則顯著高於資料庫管理人員。這樣的結果與 Mart´ın et al.(2018)的研究結果相似，其針對於西班牙的資訊軟體系統類相關職缺的研究，也發現職位對於薪資水準有顯著之影響。在第五章，則會透過建構薪資預測模型，分析職位的差異對於薪資的影響。

DOI:10.6814/NCCU202001406

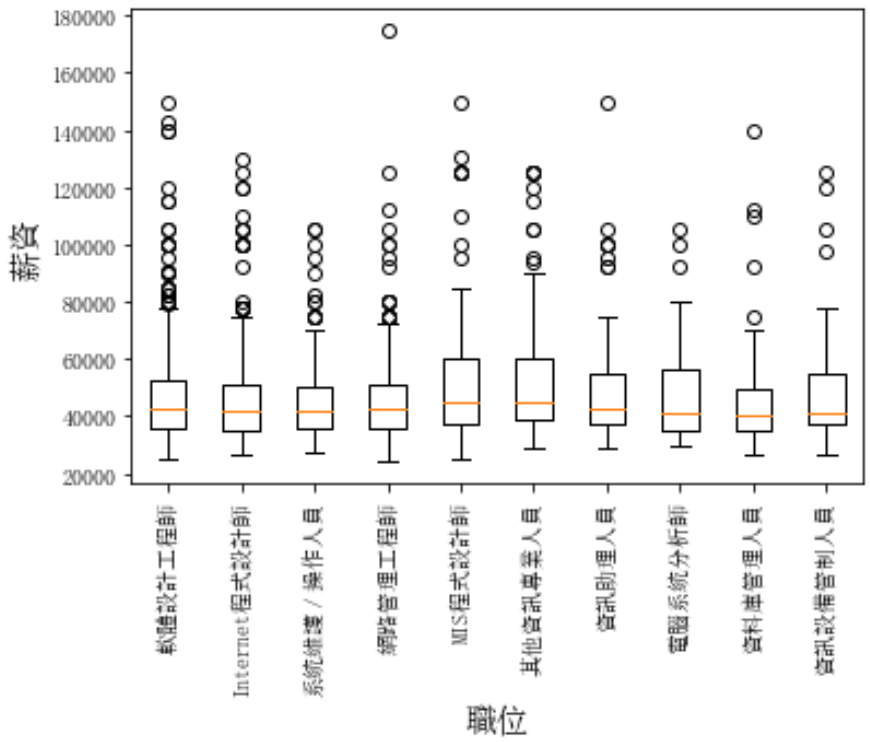

圖 4-33 職位與薪資箱型圖 (圖中的橘線為中位數) (註: 圖中擷取在資料集中出現較多的職位，由左至右出現的頻率為遞減)

第三節 工作內容文字前處理

除了人力銀行上的結構化變數，本研究也嘗試將工作內容的文字描述，轉換成預測模型之變數，以增加預測效能。首先，以正規表達式去除工作描述內容中的非語言成分，例如一般的標點符號與 41 個英文停用詞，接著以 jieba 中文斷詞套件使用精確模式斷詞，最後分別將斷詞後的文字轉換成 TF 以及 TF-IDF，作為預測薪資之變數。同時，斷詞後的工作內容也使用 Keras 套件的嵌入層，將其轉換成以詞向量表示的矩陣。在過程中也嘗試運用中文維基百科的預訓練詞向量作為訓練的基礎，再透過 Keras 套件的嵌入層訓練與調整，如此之遷移學習對於訓練我們的文本詞向量，期望能有提升預測效能與時間效率的功用。

DOI:10.6814/NCCU202001406

此外，也將工作內容的文字描述長度當作一個變數，由圖 4-34 可見文字長度幾乎都落在 500 字元以下，但同時也存在許多較多字元的離群值。

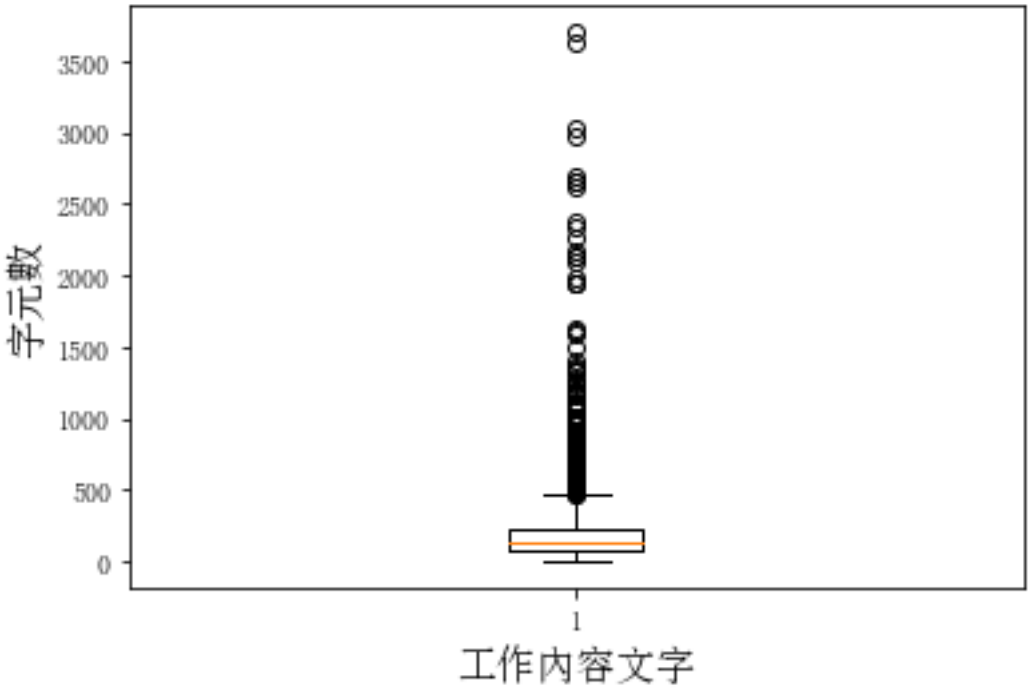

圖 4-34 工作內容文字長度分布箱型圖 (圖中的橘線為中位數)

由圖 4-35 可以發現，工作內容文字的長度與薪資水準為正向相關，且相關係數為 0.378，具有一定之預測能力，故將其納入變數。此外，為了使此一連續變數之尺度與其他獨熱編碼(One-hot encoding)之類別變數相近，在此使用標準化:原值減去平均數並除以標準差的方式限縮變數之尺度。

DOI:10.6814/NCCU202001406

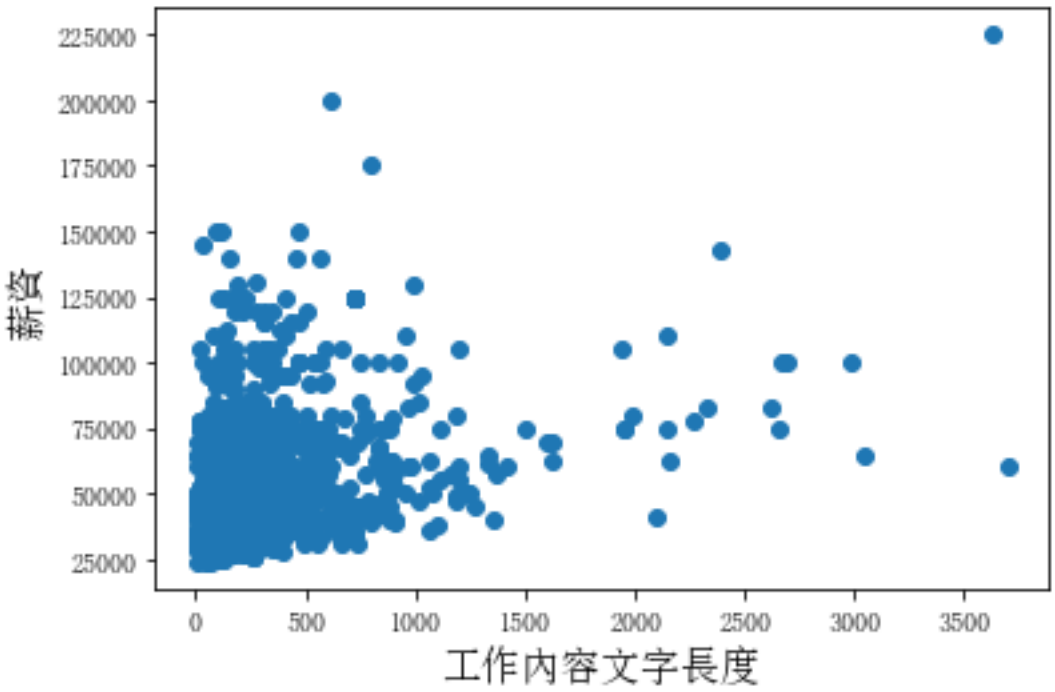

圖 4-35 工作內容文字長度與薪資散佈圖

第四節 資料總結

經資料前處理後，資料筆數從原本的 2935 筆縮減為 2633 筆，刪除的資料筆數為工作案件薪資較不明確者，如以年薪、時薪計算或是範圍較含糊，共計 302 筆資料被刪除。最後的變數名稱與轉換後的變數維度羅列於表 4-4，經過資料前處理的轉換過程，原本的 12 個類別變數藉由獨熱編碼轉換為 756 維度的資料表。其中表 4-4 中轉換後變數維度的欄位是指在資料表中該變數實際使用的欄位數量，是變數類別數量減一，因為在獨熱編碼轉換刪除第一個類別以避免共線性。另外，工作內容的文字描述也考慮先去除英文的停用詞，並轉換成 TF 與 TF-IDF 以及使用 Keras 套件的嵌入層轉換而得詞向量。其中，在表 4-5 的 TF 與 TF-IDF 的轉換後變數維度為 13951，是因為 13951 即為資料集中所有字詞的總數，而詞向量則是(1290,300)的矩陣，其中 1290 代表最長文本的字詞數量，並將每個字詞以 300 個維度的向量型態表示。我們將會使用這些轉換後的變數訓練薪資預測模型，詳細的內容與研究結果將於下一章呈現。

DOI:10.6814/NCCU202001406

表 4-4 前處理後結構化變數資料總結表

變數名稱	原始資料型態	轉換後變數維度
薪資	連續	1
地區	類別	21
管理責任	類別	1
出差要求	類別	5
上班時間	類別	1
周休	類別	1
經歷	類別	9
學歷	類別	5
科系	類別	68
語言	類別	2
擅長工具	類別	299
工作技能	類別	184
職位	類別	160

表 4-5 資料前處理後工作內容文字資料總結表

變數名稱	原始資料型態	轉換後變數維度
工作內容文字長度	連續	1
TF	向量	13951
TF-IDF	向量	13951
詞向量	矩陣	(1290,300)

DOI:10.6814/NCCU202001406

第五章 迴歸模型與實證結果

本章節旨在說明薪資預測模型的建構過程與結果。第一節以薪資本身的平均做預測，以作為後續迴歸模型的基準模型。第二節主要討論變數組合生成的過程。第三節則交叉比對不同的變數組合與迴歸模型的評估表現。第四節針對挑選出來的變數組合與模型，檢視變數之重要性，進而篩選出對於薪資水準有顯著影響之變數。最後第五節則嘗試使用詞向量轉換作為變數，進而建構薪資預測模型，主要使用的模型架構為 TextCNN，並針對許多神經網路的超參數進行調整，例如卷積層、池化層之超參數，並基於這些超參數去建立一個卷積神經網路輸出薪資預測。

第一節 基準模型

我們不使用任何工作案件中的資訊建立基準模型，該模型僅使用所有已知薪資的平均值來預測其餘工作案件的薪資，我們在這簡單的基準模型中使用 5 則交叉驗證，其評估結果如表 5-1 所示。接著，我們將使用工作案件中的相關資訊建構薪資預測模型，他們的結果將與該基準模型比較。

表 5-1 基準模型評估結果表

判定係數	平均平方誤差	平均絕對誤差
-0.006	342271303	12687

第二節 變數組合生成

將斷詞後的工作內容描述的文字資料，透過 TF 和 TF-IDF 分別轉換為兩個具有 13951 維度的資料表，並分別與先前的結構化變數合併，產生共計五種之變數組合。顯然，5 個數據集都具有非常高的維數，除了 x 之外，其餘的變數組合的維度甚至比資料

DOI:10.6814/NCCU202001406

筆數還大。這使得模型建構成為一項艱鉅的任務，我們將對這些高維數據集，應用各種線性和非線性迴歸模型做薪資預測。

表 5-2 變數組合概要表

簡稱	結構化變數	工作內容文字 TF	工作內容文字 TF-IDF	維度
x	V			757
tf		V		13951
idf			V	13951
x_tf	V	V		14708
x_idf	V		V	14708

第三節 變數組合與迴歸模型篩選

將 5 種不同的變數組合搭配 11 種迴歸模型，並做 5 則交叉驗證。我們的評估結果顯示，與基準模型相比，所有 3 個評估標準(判定係數，平均平方誤差和平均絕對誤差)，在所有的迴歸模型均具有更好的預測準確性，這顯示工作案件中的資訊具有薪資的預測能力。表 5-3 列出了所有 5 種數據集的 3 種最佳模型的預測結果，所有變數組合的預測結果前三名分別是隨機森林、Ridge、Lasso。在隨機森林中，決策樹的建構是基於隨機選擇的子樣本和部分的變數，換句話說，隨機森林並非將所有的高維特徵都用於樹的建構。為了避免在決策樹建構的過程中忽略某些變數，我們使用整個資料集的變數，也就是預設的超參數設定完成樹的分枝，結果顯示這在高維特徵資料集上效果很好(Breiman, 2001)。Ridge 和 Lasso 在高維和稀疏資料上的表現也相當不錯(Friedman et al., 2010)，我們的預測結果顯示，這兩種算法始終在所有 5 個數據集中位居前 3 名。在所有的預測模型與變數組合中，最佳的評估結果為隨機森林迴歸搭配 x_idf，其平均絕對

DOI:10.6814/NCCU202001406

誤差是 8706 元。但是，以上所有模型的判定係數都小於 0.5，顯示這些模型仍有可以改進的空間，我們將在第六章中概述可以改進的地方。

表 5-3 變數組合與迴歸模型預測結果

迴歸模型	變數組合	判定係數(R^2)	平均平方誤差(MSE)	平均絕對誤差(MAE)
Ridge	x	0.45	188988513	9390
Lasso	x	0.42	196862613	9519
隨機森林	x	0.41	200889147	8970
隨機森林	corpustf	0.25	255156463	9843
Ridge	corpustf	0.24	259063835	10710
Lasso	corpustf	0.22	267780287	10776
Ridge	corpusidf	0.33	227698973	9838
隨機森林	corpusidf	0.26	253616647	9864
Lasso	corpusidf	0.25	255882990	10357
Ridge	x_idf	0.47	180134295	9070
Lasso	x_idf	0.44	191520526	9290
隨機森林	x_idf	0.41	202876357	8706
Ridge	x_tf	0.45	188197831	9362
Lasso	x_tf	0.42	196666048	9518
隨機森林	x_tf	0.39	208761575	8782

從表 5-3 的評估結果可以發現，在所有三種迴歸演算法，都有 MAE（x_idf）<MAE（x_tf）<MAE（x）<MAE（idf）<MAE（tf）的評估結果。這顯示在三種變數中(結構化變數、TF、TF-IDF)，結構化變數包含最多的薪水攸關資訊，其次是 TF-IDF，最後才

DOI:10.6814/NCCU202001406

是 TF。這樣的結果實屬合理，因為結構化變數比非結構化的文本對於工作職位的相關條件要求明顯更為精確。同時，單詞在文本中出現的頻率較高（高 TF）並不意味著該單詞很重要，除非它在該資料集的其他文本中出現的次數相對較少（高 TF-IDF）。此外，非結構化變數和結構化變數的組合比單獨使用兩種類型的數據提供了更好的模型評估結果，這顯示兩種類型的數據都有包含薪資信息，可為薪資預測所用。

第四節 變數篩選與顯著性

接著我們使用在第三節中得到的最佳迴歸模型與變數組合，也就是隨機森林迴歸搭配 x_idf 得出變數之重要性，並依變數重要性排序。首先，比較變數個數與平均絕對誤差的關係，衡量需要使用的變數數量，如圖 5-1 所示，自擷取變數個數在 700 個之後的平均絕對誤差便沒有顯著下降，因此在往後的模型僅使用這 700 個重要性最高的變數。使用這 700 個變數在 5 則交叉驗證的情況下，其平均絕對誤差為 8659 元，在該測試集的平均薪資為 47346 元的情況下，測試集的平均絕對誤差比例約為 18.3%。

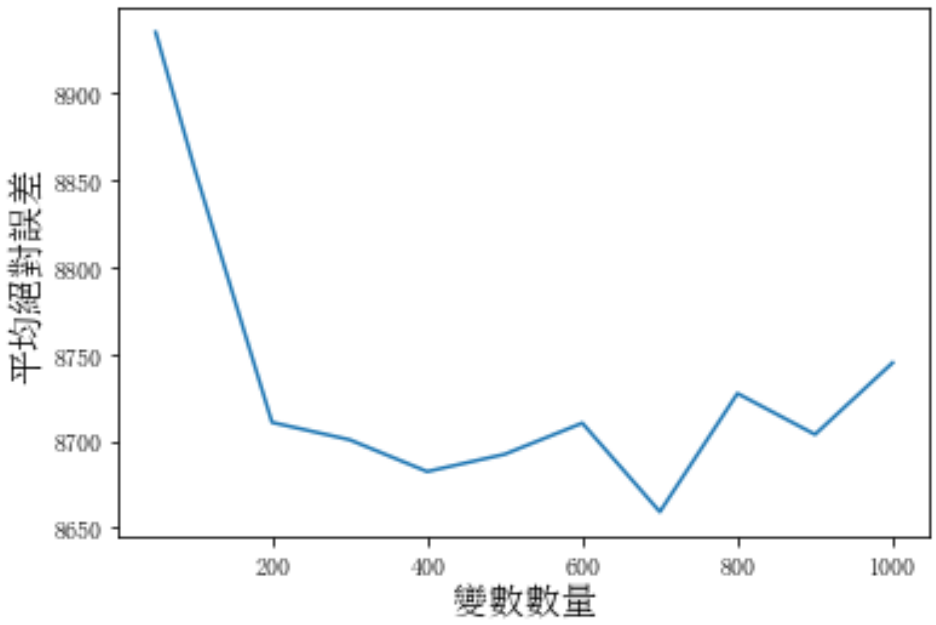

圖 5-1 平均絕對誤差與變數數量關係圖

DOI:10.6814/NCCU202001406

因為隨機森林迴歸使用樹狀結構，所以無法了解變數對於薪資的影響大小，因此嘗試使用 Ridge 演算法搭配 x_idf[26]建立模型並輸出前一百名的重要變數如表 5-4 所示，以便參考變數對於薪資水準的影響。結果與我們之前進行的資料探索性分析相似，以工作地區為例，在國外的工作案件如東南亞或日本，其對於薪資預測模型都具有相當顯著之影響。另外，我們也可以從該表中發現經歷、需負擔管理責任與長時間出差的工作案件，也對於薪資預測模型具有相當顯著之影響。同時，我們也能夠從擅長工具、技能、職位的子表中了解哪些擅長工具、技能、職位具有比較高的市場價值，例如 Spring Framwork 的知識、軟體品質與保證、軟體專案主管等等對於薪資水準的影響。最後，我們也可以從 TF-IDF 的字詞中了解，通常在工作內容的文字描述中提到哪些字詞的工作案件，其薪資水準會有甚麼樣的趨勢與變化，同時，這些字詞也顯示員工在工作中某些行為對薪資之影響例如協助、執行。

表 5-4 Ridge 重要變數表(以係數絕對值排序並只取前一百名再以變數類別分割成子表)

排名	工作地區
1	address_東南亞
3	address_日本
11	address_台南
17	address_彰化
24	address_高雄
25	address_台中
39	address_桃園

[26] 在 Ridge 演算法選擇使用 x_idf 作為輸出係數的變數組合，是因為其在薪資預測模型的評估結果為所有變數組合之中最好的。

DOI:10.6814/NCCU202001406

排名	經歷
7	exp_不拘
8	exp_1 年以上
10	exp_5 年以上
41	exp_8 年以上
50	exp_2 年以上

排名	管理責任
30	manager_需負擔管理責任

排名	出差要求
2	trip_Travel < 7 months/year

排名	擅長工具
4	Spring_須會
12	Shell_須會
14	Scala_須會
52	J2SE_須會
66	ANSI SQL_須會
67	C#_須會
70	iptables_須會
76	Struts_須會
85	ReactJS_須會
87	Python_須會
94	JSP_須會

DOI:10.6814/NCCU202001406

96	C++.Net_須會

排名	技能
16	軟體品質與保證_具有
31	使用者測試(Usability test)_具有
33	網路設備設定安裝_具有
34	軟硬體設備成本控制_具有
44	無線通訊技術開發_具有
49	資料備份與復原_具有
56	網路規劃管理_具有
60	網路系統配置_具有
61	網路應用軟體操作_具有
75	通訊工程技術開發_具有
81	測試計劃及測試報告書撰寫_具有
82	安裝與維護網路安全系統_具有
84	設計網路安全系統_具有
91	測試環境建置規劃_具有
95	規劃與管理網路入侵檢測系統（NIDS）_具有
98	軟體工程系統開發_具有

排名	職位
5	軟體專案主管_是
9	電玩程式設計師_是
15	市場調查／市場分析_是
18	電子商務技術主管_是

DOI:10.6814/NCCU202001406

21	軟體設計工程師_是
22	資訊助理人員_是
27	網路安全分析師_是
28	演算法開發工程師_是
40	業務助理_是
53	助理工程師_是
71	品管／檢驗人員_是
73	通訊軟體工程師_是
80	資料輸入人員_是
92	產品維修人員_是

排名	**TF-IDF**
6	性能
13	精通
19	sdk
20	程式
23	協助
26	調優
29	資訊
32	電腦
35	優化
36	服務器
37	相關
38	以上

DOI:10.6814/NCCU202001406

42	sql
43	要會馬甲包
45	ios
46	plus
47	作業
48	技術
51	客戶
54	工作
55	網路
57	length
58	定位
59	下班
62	人才
63	配合
64	架構
65	系统
68	大型
69	app
72	redis
74	效能
77	維護
78	研發
79	訓練
83	優先

DOI:10.6814/NCCU202001406

86	popdaily
88	腳本
89	行銷
90	安全
93	基本
97	執行
99	一定
100	互聯

第五節 詞向量轉換薪資預測模型建構

本節嘗試使用卷積神經網路做為詞向量轉換的預測模型，本研究的神經網路架構主要使用 TextCNN。在 TextCNN 的模型中，我們需要在嵌入層、卷積層和池化層調整參數，我們使用 80-20 的訓練與驗證數據進行網格搜尋，在每一個參數組合中都讓神經網路訓練 10 個 epochs，並選擇具有最佳驗證結果的設定，詳細的調整目標與參數設定如表 5-5 所示。其中，嵌入層方式即是 3.4.3.2 所述的三種嵌入層權重的輸入方式，過濾層數量即為在卷積過程中欲使用的特徵轉換圖數量，步伐大小為卷積過程中每次過濾層所移動的距離，池化大小則為池化過程中，要將多大範圍的特徵視為一組進行池化。

表 5-5 詞向量轉換神經網路參數調整表

調整目標	**參數設定**
嵌入層方式	(CNN-rand, CNN-non-static, CNN-static)
過濾層數量	(8,16,32,64,128)
步伐大小	(1,2,3,4,5)
池化大小	(2,3,4,5,6)

DOI:10.6814/NCCU202001406

實驗結果發現[27]，(嵌入層方式:CNN-rand, 過濾層數量:128, 步伐大小:1, 池化大小:3)的組合可以在驗證集得到最小的平均絕對誤差，其驗證集的平均絕對誤差可以達到 8656 元，在平均薪資為 47346 元的情況下，其平均絕對誤差比例可以達到 18.3%。

[27] 完整的實驗結果如附錄表六所示

DOI:10.6814/NCCU202001406

第六章 結論與建議

第一節 結論

本研究使用 104 人力銀行的資訊軟體系統類的工作案件資料，除了結構化的應徵者變數與工作的相關變數，也包含了工作內容描述的文字轉換，經過資料前處理，建立薪資預測模型，作為企業方與求職者的參考依據，在嘗試了多種迴歸模型與變數組合的搭配之後，依照平均絕對誤差的評估準則，我們發現使用隨機森林搭配 x_idf 迴歸建立模型可以得到最佳的薪資預測，其在測試集的平均絕對誤差比例可以達到 18.3%。同時，我們也以 Ridge 搭配 x_idf 輸出模型的變數係數，了解各項變數對於薪資水準的影響，對於求職者而言，可以直接明白各項職場能力的市場價值，作為他們在未來自我精進的參考依據。

此外，本研究也嘗試只使用工作內容描述的文字，以詞向量的變數轉換方式，在 TextCNN 的架構下建立薪資預測模型，同時也嘗試超參數調整，最後在驗證集的平均絕對誤差比例可以達到 18.3%，在缺乏結構化變數的情況下，可以提供給求職者作為薪資預測之工具。

第二節 建議

雖然本研究的的薪資預測模型架構，在台灣地區的資訊軟體系統類相關工作案件的資料集上有不錯的評估結果，但該預測模型仍然有改進的空間，我們將在這節中概述可以改進的地方。

一、資料層面

DOI:10.6814/NCCU202001406

首先，使用網路人力銀行案件做為資料來源有其樣本代表性問題，包括實際有多少企業會使用該平台作為徵才媒介，以及案件中的結構化變數是否與所有的徵才案件有所偏差，因此本研究之解釋能力僅止於網路人力銀行徵才案件的工作範疇，而為了進一步縮小樣本變異，此研究進一步將範圍限縮在資訊軟體系統類的工作職缺，又再將模型解釋範圍進一步限縮，因此未來的研究可依據欲解釋的面向與範疇選擇資料來源，並留意模型能夠解釋的範圍。

再者，單方面透過網路人力銀行徵才案件建構模型，可能會產生資料偏誤的問題，最明顯的例子表現在薪資層面上。實際薪資大於四萬的工作職缺往往被標記為面議，亦或是薪資資訊呈現的是一個區間範圍，實際的薪資水準通常會在面試時依據求職者的個人背景做調整，因此直接將面議的工作案件剔除，或是使用平均值作為區間薪資的最終輸出，都有可能直接造成薪資訓練標籤的偏誤。可能的解決方案為尋找其它資料來源(若研究標的不限於網路人力銀行)，或是針對較低薪族群建立薪資預測模型。

在資料前處理的部分，本研究簡化了部分資料的類別項目，例如在語言要求方面只簡單地分成三類，而不是為每一種語言標記一個類別，這樣的作法並不一定是一個問題，將個數較少的類別結合有助於模型的一般化。例如，若求職者具備網路人力銀行上沒有註明的語言能力，那他就可以直接參考其他語言的類別，同時這類做法也能有效降低變異，限縮部分變數的離群值對於模型的影響力，也可以解決資料筆數不足時的統計顯著性問題。相反地，缺點即是類別可能不夠明確，較大的分類群體可能與單一類別的分布有著極大的差異，如此的做法便可能直接導致預測的偏誤。綜上所述，實際的資料處理層面仍要依據該資料集的特性做處理，是否要將數量較少的類別項目整合，則並沒有一定的答案。

DOI:10.6814/NCCU202001406

在中文斷詞的部分使用 jieba 套件，該套件雖然已日趨成熟，但還是可能會有部分字詞切分不明確的狀況發生，此時若可以在自定義詞典加入相關的中文工作類的專業用詞，期待將能有效地改進斷詞結果。

二、模型層面

首先，若使用工作內容的文字作為變數，因為部分字詞出現在資料集裡面的頻率太低，可能會讓模型變得較不穩健，因此可以考慮去除出現頻率過低的字詞變數，或甚至是以人工的方式挑出工作內容中的關鍵字詞，再一一比對各案件中出現的字詞。

另外，關於詞向量轉換的模型建構，這裡直接使用 TextCNN 的架構，並針對相關細節進行實驗，當然在此也可以考慮調整更多細節，或甚至是直接重新建立一個卷積層架構進行實驗。同時，這裡使用中文維基百科的預訓練詞向量作為初始權重，未來的研究者也可以考慮使用其他預訓練詞向量集，或甚至是使用 Word2vec 針對自定義文本進行訓練。

DOI:10.6814/NCCU202001406

參考文獻

[1] 104 人力銀行，AI 大浪捲動企業搶才職缺是 5 年前的 3.2 倍，上網日期 2020 年 06 月 20 日，檢自：https://corp.104.com.tw/archive/files/news/20200121.pdf

[2] 104 人力銀行，上網日期 2020 年 06 月 20 日，檢自：https://www.104.com.tw/jobs/main/https://www.cnbc.com/2019/12/30/5-hig

[3] Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, *46*(3), 175-185.

[4] Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

[5] Breiman, L., J. Friedman, R. Olshen, and C. Stone, (1984). Classification and Regression Trees. Belmont, California : Wadsworth International Group.

[6] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of machine learning research*, *12*(ARTICLE), 2493-2537.

[7] Drucker, H., Burges, C.J.C., Kaufman, L., Smola, A. and Vapnik, V, (1997). “Support vector regression machines”, Advances in Neural Information Processing Systems, 9:155–161.

[8] Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, *55*(1), 119-139.

[9] Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, *33*(1), 1.

[10] Hinton, G. E. (1990). Connectionist learning procedures. In *Machine learning* (pp. 555-610). Morgan Kaufmann.

DOI:10.6814/NCCU202001406

[11] Jones, K. S. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*.

[12] Keras, Retrieved June 20 2020, from: https://keras.io/

[13] Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

[14] Martín, I., Mariello, A., Battiti, R., & Hernández, J. A. (2018). Salary Prediction in the IT Job Market with Few High-Dimensional Samples: A Spanish Case Study. *International Journal of Computational Intelligence Systems*, *11*(1), 1192-1209.

[15] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

[16] Pawha, A., & Kamthania, D. (2019). Quantitative analysis of historical data for prediction of job salary in India-A case study. *Journal of Statistics and Management Systems*, *22*(2), 187-198.

[17] Scikit-learn, Retrieved June 20 2020, from: https://scikit-learn.org/stable/

[18] Selenium with Python, Retrieved June 20 2020, from: https://selenium-python.readthedocs.io/

[19] Singh, R. (2016). A Regression Study of Salary Determinants in Indian Job Markets for Entry Level Engineering Graduates.

[20] Sun Junyi，结巴中文分词，上網日期 2020 年 06 月 20 日，檢自 https://github.com/fxsjy/jieba

[21] Support Vector Machine - Regression(SVR), Retrieved June 20 2020, from: http://www.saedsayad.com/support_vector_machine_reg.htm

[22] These 5 high-paying, growing jobs didn't exist a decade ago—but they'll be booming through the 2020s, Retrieved June 20 2020, from: https://www.cnbc.com/2019/12/30/5-

DOI:10.6814/NCCU202001406

high-paying-growing-jobs-that-will-be-booming-through-the-2020s.html?fbclid=IwAR1mOcFVDUNxaGk5EAsbkxLU2wP40yxLb8cBqNGjrccXgXoCoiuR4_LxTTQ

[23] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267-288.

[24] Vapnik, V. N. (1995). Constructing learning algorithms. In *The nature of statistical learning theory* (pp. 119-166). Springer, New York, NY.

[25] Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, *67*(2), 301-320.

[26] 中央研究院詞庫小組，中文斷詞系統，上網日期 2020 年 06 月 20 日，檢自：http://ckipsvr.iis.sinica.edu.tw/

[27] 江易儻，(2018)。應用雙向長短期記憶神經網路於新聞分類。未出版之碩士論文，國立雲林科技大學，資訊管理系，雲林縣。

[28] 周宜滿，(2004)。高等教育薪資所得差異之經濟分析-臺灣實證研究。未出版之碩士論文，佛光大學，經濟學研究所，宜蘭縣。

[29] 林鼎晃，(2012)。大學科系別薪資決定因素分析－ 熱門科系是否代表「錢」景看好？。未出版之碩士論文，國立東華大學，經濟學系，花蓮縣。

[30] 徐豪，(2019)。使用深度學習進行基於社群網路評論的產品評價系統。未出版之碩士論文，淡江大學，資訊工程學系碩士在職專班，新北市。

[31] 莊惠婉，(2010)。影響我國產業別員工薪資之因素－應用最大概似法及兩階段有序機率選擇模型。未出版之碩士論文，國立中正大學，國際經濟研究所，嘉義縣。

DOI:10.6814/NCCU202001406

[32] 創市際市場研究顧問公司，就業調查與就業服務/職涯類別網域使用概況，上網日期 2020 年 06 月 20 日，檢自：https://www.ixresearch.com/wp-content/uploads/report/InsightXplorer%20Biweekly%20Report_20160815.pdf

[33] 曾厚強、洪孝宗、宋曜廷、陳柏琳，(2016)。基於深層類神經網路及表示學習技術之文件可讀性分類。The 2016 Conference on Computational Linguistics and Speech Processing ROCLING, pp. 255-270。

[34] 劉姿君，(1993)。教育投資與薪資報酬─人力資本理論之應用。未出版之碩士論文，國立政治大學，教育學研究所，台北市。

DOI:10.6814/NCCU202001406

附錄

附錄表一 依地區劃分的薪資 Mann-Whitney rank 檢定 p 值表(小於 0.01 以粗體表示)

地區	台北	台中	新北	桃園	高雄	新竹	台南	彰化	東南亞	苗栗	中國
台北	1.000	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.002**	**0.000**
台中	**0.000**	1.000	**0.001**	**0.001**	0.198	**0.005**	0.011	**0.007**	**0.000**	0.224	**0.000**
新北	**0.000**	**0.001**	1.000	0.345	**0.000**	0.478	**0.000**	**0.000**	**0.000**	0.027	**0.000**
桃園	**0.000**	**0.001**	0.345	1.000	**0.000**	0.344	**0.000**	**0.000**	**0.000**	0.027	**0.000**
高雄	**0.000**	0.198	**0.000**	**0.000**	1.000	**0.001**	0.064	0.014	**0.000**	0.381	**0.000**
新竹	**0.000**	**0.005**	0.478	0.344	**0.001**	1.000	**0.000**	**0.000**	**0.000**	**0.027**	**0.000**
台南	**0.000**	0.011	**0.000**	**0.000**	0.064	**0.000**	1.000	0.095	**0.000**	0.360	**0.000**
彰化	**0.000**	**0.007**	**0.000**	**0.000**	0.014	**0.000**	0.095	1.000	**0.000**	0.146	**0.000**
東南亞	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	1.000	**0.000**	0.073
苗栗	**0.002**	0.224	0.027	0.027	0.381	0.027	0.360	0.146	**0.000**	1.000	**0.000**
中國	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.073	**0.000**	1.000
宜蘭	**0.000**	0.022	**0.003**	**0.003**	0.028	**0.002**	0.069	0.238	**0.000**	0.051	**0.000**
南投	**0.004**	0.173	0.034	0.029	0.267	0.028	0.483	0.312	**0.000**	0.275	**0.000**
屏東	0.031	0.413	0.152	0.106	0.433	0.177	0.292	0.182	**0.000**	0.289	**0.000**
雲林	**0.003**	0.074	0.014	0.021	0.054	**0.009**	0.101	0.455	**0.000**	0.080	**0.000**
嘉義	**0.000**	**0.001**	**0.000**	**0.000**	**0.001**	**0.000**	**0.002**	0.015	**0.000**	**0.006**	**0.000**
日本	**0.004**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.093	**0.000**	**0.004**
台東	0.022	0.287	0.082	0.094	0.345	0.066	0.405	0.154	**0.000**	0.375	**0.001**
花蓮	0.024	0.217	0.076	0.068	0.203	0.057	0.323	0.384	**0.000**	0.277	**0.002**

DOI:10.6814/NCCU202001406

基隆	0.028	0.215	0.089	0.071	0.253	0.071	0.372	0.377	**0.001**	0.342	**0.003**
金門	0.274	0.487	0.412	0.361	0.410	0.401	0.319	0.244	0.076	0.348	0.078
歐洲	0.093	0.047	0.057	0.048	0.047	0.051	0.044	0.050	0.348	0.059	0.078

附錄表二 依地區劃分的薪資 Mann-Whitney rank 檢定 p 值表 (小於 0.01 以粗體表示)

地區	宜蘭	南投	屏東	雲林	嘉義	日本	台東	花蓮	基隆	金門	歐洲
台北	**0.000**	**0.004**	0.031	**0.003**	**0.000**	**0.004**	0.022	0.024	0.028	0.274	0.093
台中	0.022	0.173	0.413	0.074	**0.001**	**0.000**	0.287	0.217	0.215	0.487	0.047
新北	**0.003**	0.034	0.152	0.014	**0.000**	**0.000**	0.082	0.076	0.089	0.412	0.057
桃園	**0.003**	0.029	0.106	**0.021**	**0.000**	**0.000**	0.094	0.068	0.071	0.361	0.048
高雄	0.028	0.267	0.433	0.054	**0.001**	**0.000**	0.345	0.203	0.253	0.410	0.047
新竹	**0.002**	0.028	0.177	**0.009**	**0.000**	**0.000**	0.066	0.057	0.071	0.401	0.051
台南	0.069	0.483	0.292	0.101	**0.002**	**0.000**	0.405	0.323	0.372	0.319	0.044
彰化	0.238	0.312	0.182	0.455	0.015	**0.000**	0.154	0.384	0.377	0.244	0.050
東南亞	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.093	**0.000**	**0.000**	**0.001**	0.076	0.348
苗栗	0.051	0.275	0.289	0.080	**0.006**	**0.000**	0.375	0.277	0.342	0.348	0.059
中國	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.004**	**0.001**	**0.002**	**0.003**	0.078	0.078
宜蘭	1.000	0.225	0.123	0.442	0.068	**0.001**	0.076	0.230	0.292	0.191	0.081
南投	0.225	1.000	0.385	0.131	0.011	**0.001**	0.452	0.473	0.348	0.430	0.079
屏東	0.123	0.385	1.000	0.184	0.011	**0.001**	0.371	0.357	0.271	0.421	0.085
雲林	0.442	0.131	0.184	1.000	0.052	**0.006**	**0.035**	0.304	0.247	0.164	0.086
嘉義	0.068	0.011	0.011	0.052	1.000	**0.002**	0.010	0.044	0.052	0.136	0.095
日本	**0.001**	**0.001**	**0.001**	**0.006**	**0.002**	1.000	**0.002**	**0.004**	**0.007**	0.106	0.401
台東	0.076	0.452	0.371	0.035	0.010	**0.002**	1.000	0.200	0.289	0.288	0.084

DOI:10.6814/NCCU202001406

花蓮	0.230	0.473	0.357	0.304	0.044	**0.004**	0.200	1.000	0.500	0.279	0.121
基隆	0.292	0.348	0.271	0.247	0.052	**0.007**	0.289	0.500	1.000	0.346	0.138
金門	0.191	0.430	0.421	0.164	0.136	0.106	0.288	0.279	0.346	1.000	0.500
歐洲	0.081	0.079	0.085	0.086	0.095	0.401	0.084	0.121	0.138	0.500	1.000

附錄表三 依出差要求劃分的薪資 Mann-Whitney rank 檢定 p 值表(小於 0.01 以粗體表示)

出差要求	不需出差	出差時間未定	出差小於一個月/年	出差小於三個月/年	出差小於七個月/年	出差小於六個月/年
不需出差	1.000	0.280	0.287	0.014	0.010	0.136
出差時間未定	0.280	1.000	0.432	0.033	0.013	0.131
出差小於一個月/年	0.287	0.432	1.000	0.039	0.015	0.111
出差小於三個月/年	0.014	0.033	0.039	1.000	0.037	0.027
出差小於七個月/年	0.010	0.013	0.015	0.037	1.000	0.019
出差小於六個月/年	0.136	0.131	0.111	0.027	0.019	1.000

附錄表四 依經歷劃分的薪資 Mann-Whitney rank 檢定 p 值表(小於 0.01 以粗體表示)

經歷	1 年以上	2 年以上	3 年以上	4 年以上	5 年以上	6 年以上	7 年以上	8 年以上	10 年以上
1 年以上	1.000	**0.000**	**0.000**	**0.000**	**0.000**	0.240	0.043	**0.001**	0.042
2 年以上	**0.000**	1.000	**0.000**	**0.000**	**0.000**	0.102	0.048	**0.003**	0.044
3 年以上	**0.000**	**0.000**	1.000	0.039	**0.007**	0.070	0.089	0.037	0.051
4 年以上	**0.000**	**0.000**	0.039	1.000	0.465	0.057	0.066	0.103	0.057
5 年以上	**0.000**	**0.000**	**0.007**	0.465	1.000	0.055	0.085	0.129	0.060
6 年以上	0.240	0.102	0.070	0.057	0.055	1.000	0.500	0.144	0.500
7 年以上	0.043	0.048	0.089	0.066	0.085	0.500	1.000	0.362	0.500

DOI:10.6814/NCCU202001406

8 年以上	**0.001**	**0.003**	0.037	0.103	0.129	0.144	0.362	1.000	0.362
10 年以上	0.042	0.044	0.051	0.057	0.060	0.500	0.500	0.362	1.000

附錄表五 依學歷劃分的薪資 Mann-Whitney rank 檢定 p 值表(小於 0.01 以粗體表示)

學歷	高中	專科	大學	碩士	博士
高中	1.000	0.013	**0.000**	**0.000**	0.026
專科	0.013	1.000	**0.000**	**0.000**	**0.009**
大學	**0.000**	**0.000**	1.000	**0.004**	0.015
碩士	**0.000**	**0.000**	**0.004**	1.000	0.030
博士	0.026	**0.009**	0.015	0.030	1.000

附錄表六 詞向量轉換神經網路參數調整驗證集平均絕對誤差表(依平均絕對誤差排序)

嵌入層方式	過濾層數量	步伐大小	池化大小	平均絕對誤差
CNN-rand	128	1	3	8656
CNN-rand	128	1	2	8756
CNN-rand	128	2	2	8757
CNN-rand	128	3	2	8770
CNN-non-static	128	1	3	8799
CNN-rand	64	1	2	8815
CNN-rand	64	1	3	8859
CNN-rand	128	1	4	8865
CNN-rand	128	1	5	8871
CNN-rand	32	1	2	8880
CNN-non-static	128	1	2	8883
CNN-rand	64	2	2	8884
CNN-non-static	128	2	2	8917
CNN-rand	128	2	3	8918
CNN-rand	128	4	2	8933
CNN-rand	32	1	3	8950
CNN-non-static	64	1	2	8950
CNN-rand	128	2	4	8986

DOI:10.6814/NCCU202001406

CNN-non-static	128	1	6	8988
CNN-rand	128	1	6	8989
CNN-rand	32	2	2	9010
CNN-non-static	128	2	3	9017
CNN-rand	64	2	3	9018
CNN-rand	128	3	3	9054
CNN-non-static	64	1	3	9055
CNN-rand	64	1	4	9067
CNN-non-static	32	1	2	9077
CNN-rand	32	1	4	9079
CNN-rand	128	5	2	9092
CNN-rand	128	3	5	9098
CNN-non-static	32	1	3	9101
CNN-rand	64	1	6	9107
CNN-rand	64	3	2	9108
CNN-rand	64	1	5	9126
CNN-non-static	128	3	2	9129
CNN-non-static	128	4	2	9132
CNN-non-static	128	3	3	9133
CNN-rand	64	4	2	9141
CNN-rand	128	3	4	9160
CNN-non-static	128	1	4	9167
CNN-rand	128	4	3	9180
CNN-rand	128	2	6	9188
CNN-non-static	128	1	5	9200
CNN-rand	128	2	5	9204
CNN-rand	32	2	3	9211
CNN-non-static	128	5	2	9213
CNN-rand	128	4	4	9224
CNN-rand	128	5	3	9225
CNN-rand	64	2	4	9229
CNN-non-static	64	1	4	9232
CNN-non-static	32	1	4	9234
CNN-rand	32	3	2	9240
CNN-rand	32	1	5	9241
CNN-non-static	128	2	4	9243

DOI:10.6814/NCCU202001406

CNN-non-static	32	2	2	9247
CNN-non-static	128	2	6	9249
CNN-non-static	64	1	5	9259
CNN-rand	32	1	6	9281
CNN-rand	128	3	6	9284
CNN-non-static	64	1	6	9295
CNN-rand	32	4	2	9298
CNN-non-static	64	3	2	9306
CNN-rand	128	5	4	9313
CNN-rand	128	4	5	9336
CNN-non-static	128	4	3	9338
CNN-non-static	128	2	5	9340
CNN-rand	64	2	5	9361
CNN-non-static	128	3	4	9370
CNN-rand	64	3	3	9385
CNN-non-static	128	3	5	9399
CNN-rand	64	5	2	9400
CNN-non-static	64	2	2	9412
CNN-rand	64	4	3	9419
CNN-rand	128	5	6	9430
CNN-rand	64	3	5	9457
CNN-rand	32	2	4	9461
CNN-rand	32	3	3	9469
CNN-non-static	32	1	5	9489
CNN-rand	128	5	5	9496
CNN-non-static	32	1	6	9501
CNN-rand	64	3	4	9502
CNN-non-static	64	4	2	9511
CNN-non-static	128	5	3	9513
CNN-rand	64	4	4	9518
CNN-non-static	128	3	6	9522
CNN-non-static	64	2	3	9524
CNN-rand	64	2	6	9532
CNN-non-static	64	2	5	9538
CNN-rand	32	2	5	9575
CNN-non-static	32	3	2	9577

DOI:10.6814/NCCU202001406

CNN-non-static	32	2	3	9588
CNN-rand	32	5	2	9589
CNN-rand	128	4	6	9602
CNN-non-static	64	3	3	9608
CNN-non-static	128	4	4	9633
CNN-rand	32	4	3	9636
CNN-non-static	32	4	2	9643
CNN-non-static	64	2	4	9647
CNN-non-static	128	4	5	9664
CNN-non-static	128	5	4	9673
CNN-non-static	64	5	2	9699
CNN-rand	64	5	3	9706
CNN-non-static	64	4	3	9732
CNN-rand	32	2	6	9789
CNN-non-static	32	2	4	9791
CNN-non-static	64	2	6	9792
CNN-rand	32	3	4	9794
CNN-non-static	32	2	5	9801
CNN-non-static	32	3	3	9828
CNN-non-static	64	3	4	9840
CNN-rand	32	3	5	9880
CNN-rand	64	3	6	9886
CNN-rand	64	5	4	9886
CNN-rand	64	4	5	9906
CNN-rand	32	4	4	9926
CNN-non-static	128	5	5	9944
CNN-non-static	128	5	6	9992
CNN-non-static	32	5	2	10021
CNN-rand	16	1	2	10041
CNN-non-static	128	4	6	10076
CNN-rand	32	3	6	10079
CNN-rand	32	5	3	10088
CNN-rand	64	5	5	10097
CNN-rand	64	4	6	10124
CNN-rand	32	4	5	10140
CNN-non-static	32	3	4	10180

DOI:10.6814/NCCU202001406

CNN-rand	16	1	3	10185
CNN-non-static	64	3	5	10199
CNN-non-static	64	5	3	10201
CNN-non-static	64	4	4	10203
CNN-rand	32	5	4	10207
CNN-static	128	1	2	10233
CNN-rand	64	5	6	10238
CNN-non-static	32	4	3	10246
CNN-non-static	64	3	6	10249
CNN-rand	16	2	2	10263
CNN-rand	16	1	4	10279
CNN-non-static	32	2	6	10283
CNN-non-static	16	1	2	10310
CNN-rand	8	1	2	10313
CNN-static	128	1	3	10321
CNN-non-static	32	4	4	10359
CNN-non-static	16	1	3	10371
CNN-non-static	16	1	4	10372
CNN-non-static	64	5	4	10387
CNN-rand	16	1	5	10387
CNN-non-static	64	4	5	10394
CNN-rand	8	1	3	10395
CNN-rand	16	3	2	10410
CNN-rand	16	1	6	10411
CNN-non-static	32	3	5	10421
CNN-non-static	32	5	3	10444
CNN-rand	32	4	6	10459
CNN-rand	16	2	3	10470
CNN-static	128	2	2	10481
CNN-rand	32	5	5	10482
CNN-non-static	8	1	2	10491
CNN-non-static	32	4	5	10495
CNN-rand	8	2	2	10505
CNN-rand	8	1	4	10516
CNN-non-static	8	1	3	10522
CNN-rand	16	4	2	10526

DOI:10.6814/NCCU202001406

CNN-static	128	1	4	10543
CNN-non-static	16	1	5	10549
CNN-non-static	64	5	5	10560
CNN-non-static	32	3	6	10564
CNN-rand	16	2	4	10572
CNN-non-static	64	4	6	10585
CNN-non-static	8	1	4	10602
CNN-non-static	16	2	2	10608
CNN-non-static	32	5	4	10622
CNN-static	128	1	5	10628
CNN-rand	8	1	5	10635
CNN-rand	32	5	6	10639
CNN-rand	16	3	3	10647
CNN-non-static	16	3	2	10671
CNN-non-static	64	5	6	10692
CNN-non-static	16	2	3	10715
CNN-non-static	8	2	2	10744
CNN-non-static	32	4	6	10753
CNN-rand	8	2	3	10767
CNN-rand	16	2	5	10775
CNN-non-static	32	5	5	10780
CNN-static	128	1	6	10797
CNN-rand	8	1	6	10801
CNN-rand	16	5	2	10804
CNN-non-static	16	1	6	10820
CNN-rand	8	3	2	10851
CNN-rand	16	4	3	10854
CNN-non-static	8	1	6	10860
CNN-non-static	8	1	5	10883
CNN-rand	16	3	4	10893
CNN-non-static	16	3	3	10903
CNN-non-static	16	4	2	10926
CNN-rand	16	2	6	10927
CNN-non-static	8	4	2	10933
CNN-rand	16	5	3	10933
CNN-rand	16	3	5	10942

DOI:10.6814/NCCU202001406

CNN-rand	8	4	2	10948
CNN-static	64	1	2	10956
CNN-static	128	3	2	10959
CNN-rand	8	3	3	10959
CNN-static	128	2	3	10967
CNN-rand	8	2	4	10976
CNN-non-static	32	5	6	11001
CNN-non-static	16	2	5	11005
CNN-non-static	16	2	4	11019
CNN-non-static	8	2	3	11043
CNN-rand	16	4	4	11070
CNN-rand	8	2	5	11072
CNN-non-static	8	2	4	11094
CNN-rand	8	5	2	11096
CNN-non-static	16	4	3	11113
CNN-non-static	8	3	3	11143
CNN-non-static	16	5	2	11146
CNN-rand	8	3	4	11148
CNN-rand	16	3	6	11155
CNN-rand	8	2	6	11157
CNN-non-static	8	3	2	11157
CNN-rand	8	4	3	11159
CNN-static	128	4	2	11169
CNN-static	128	2	4	11201
CNN-non-static	8	5	2	11214
CNN-static	128	3	3	11220
CNN-non-static	8	2	5	11251
CNN-non-static	16	2	6	11254
CNN-non-static	16	3	4	11266
CNN-rand	16	5	4	11274
CNN-rand	16	4	5	11298
CNN-rand	8	3	5	11301
CNN-non-static	8	5	3	11302
CNN-non-static	8	3	5	11306
CNN-non-static	16	4	4	11313
CNN-rand	8	5	3	11316

DOI:10.6814/NCCU202001406

CNN-static	128	2	5	11323
CNN-non-static	16	5	3	11329
CNN-non-static	16	3	5	11338
CNN-static	64	1	3	11345
CNN-static	128	5	2	11348
CNN-rand	8	4	4	11362
CNN-rand	16	5	5	11374
CNN-non-static	16	3	6	11381
CNN-non-static	8	4	3	11391
CNN-rand	16	4	6	11396
CNN-non-static	8	3	4	11440
CNN-non-static	8	2	6	11475
CNN-non-static	16	4	5	11476
CNN-rand	8	4	5	11507
CNN-rand	8	3	6	11508
CNN-non-static	16	5	4	11525
CNN-rand	8	5	4	11537
CNN-non-static	16	4	6	11546
CNN-static	64	2	2	11554
CNN-non-static	8	4	4	11560
CNN-static	32	1	2	11566
CNN-rand	16	5	6	11569
CNN-static	128	4	3	11586
CNN-non-static	8	3	6	11588
CNN-non-static	16	5	6	11597
CNN-non-static	8	5	4	11600
CNN-static	128	2	6	11609
CNN-non-static	8	4	5	11610
CNN-static	128	5	3	11617
CNN-non-static	8	4	6	11623
CNN-non-static	16	5	5	11625
CNN-rand	8	4	6	11639
CNN-static	128	3	5	11660
CNN-rand	8	5	5	11678
CNN-static	128	3	4	11692
CNN-non-static	8	5	6	11694

DOI:10.6814/NCCU202001406

CNN-static	64	1	4	11702
CNN-non-static	8	5	5	11729
CNN-static	64	1	5	11739
CNN-rand	8	5	6	11775
CNN-static	128	4	4	11810
CNN-static	128	3	6	11880
CNN-static	128	5	4	11906
CNN-static	128	4	5	11921
CNN-static	128	5	5	11923
CNN-static	32	1	4	11926
CNN-static	128	4	6	11949
CNN-static	32	1	3	11964
CNN-static	64	3	2	11975
CNN-static	64	2	3	12000
CNN-static	128	5	6	12054
CNN-static	64	1	6	12070
CNN-static	32	2	2	12155
CNN-static	32	1	5	12222
CNN-static	64	3	3	12229
CNN-static	64	4	2	12305
CNN-static	64	2	5	12367
CNN-static	64	2	4	12378
CNN-static	64	4	3	12445
CNN-static	64	3	5	12466
CNN-static	32	1	6	12474
CNN-static	64	3	4	12489
CNN-static	64	2	6	12494
CNN-static	32	2	3	12523
CNN-static	64	5	6	12542
CNN-static	64	5	2	12544
CNN-static	64	4	4	12551
CNN-static	64	5	4	12554
CNN-static	32	3	2	12555
CNN-static	64	5	5	12564
CNN-static	64	4	6	12576
CNN-static	64	4	5	12597

DOI:10.6814/NCCU202001406

CNN-static	64	5	3	12608
CNN-static	64	3	6	12643
CNN-static	16	1	2	12668
CNN-static	32	2	4	12715
CNN-static	32	5	6	12740
CNN-static	32	3	3	12785
CNN-static	32	4	2	12793
CNN-static	32	2	5	12816
CNN-static	32	5	5	12868
CNN-static	32	5	4	12876
CNN-static	32	3	4	12909
CNN-static	32	2	6	12910
CNN-static	16	1	3	12987
CNN-static	32	3	5	13025
CNN-static	32	3	6	13034
CNN-static	32	4	6	13059
CNN-static	32	4	5	13088
CNN-static	32	4	3	13099
CNN-static	32	5	2	13111
CNN-static	16	1	4	13211
CNN-static	32	5	3	13226
CNN-static	8	1	2	13250
CNN-static	32	4	4	13260
CNN-static	16	1	5	13325
CNN-static	16	2	2	13462
CNN-static	8	1	3	13489
CNN-static	16	1	6	13567
CNN-static	16	2	3	13623
CNN-static	16	3	2	13789
CNN-static	16	2	4	13814
CNN-static	8	1	4	13917
CNN-static	16	2	6	13996
CNN-static	8	1	5	14011
CNN-static	16	4	2	14039
CNN-static	8	2	2	14063
CNN-static	16	4	3	14167

DOI:10.6814/NCCU202001406

CNN-static	16	2	5	14177
CNN-static	16	3	3	14291
CNN-static	8	1	6	14297
CNN-static	16	3	4	14327
CNN-static	16	5	5	14509
CNN-static	8	3	2	14515
CNN-static	16	5	2	14540
CNN-static	16	5	4	14580
CNN-static	16	3	6	14600
CNN-static	16	4	4	14627
CNN-static	16	5	6	14700
CNN-static	16	4	6	14795
CNN-static	8	2	6	14799
CNN-static	8	2	5	14814
CNN-static	16	3	5	14838
CNN-static	16	5	3	14855
CNN-static	8	2	4	14908
CNN-static	8	5	2	14909
CNN-static	16	4	5	14939
CNN-static	8	3	3	14959
CNN-static	8	4	2	14992
CNN-static	8	2	3	15020
CNN-static	8	3	4	15443
CNN-static	8	5	3	15628
CNN-static	8	4	4	15655
CNN-static	8	4	3	15746
CNN-static	8	3	5	15844
CNN-static	8	5	5	15868
CNN-static	8	5	6	16013
CNN-static	8	4	5	16105
CNN-static	8	5	4	16211
CNN-static	8	3	6	16317
CNN-static	8	4	6	16428

DOI:10.6814/NCCU202001406