基於字典樹改善中文情緒分類效能之研究

蘇怡仁　余碩文　黃皇瑋　陳岳群
樹德科技大學資訊工程系
iansu@stu.edu.tw

摘要

近年來在社群媒體服務中的情緒分析(Sentiment Analysis)逐漸成為新興研究領域。本研究提出以貝氏分類法為主的情緒分類器 CCLM (CKIP Combined Language Model)，藉以提升將群眾意見依情緒分類時的準確度，並透過加入前綴樹(Prefix Tree)改善整體的執行效能。本實驗主要針對社群媒體平台噗浪(Plurk)中的訊息內容作中文語句的情緒分析，並觀察加入前綴樹後執行效能的改善成效。

關鍵詞：情緒分析、見解探勘、前綴樹。

Abstract

Sentiment analysis in social media service has become a new research domain in recent years. Based on the Bayes classifier, this research proposes a novel emotion classifier, CCLM (CKIP Combined Language Model), to enhance the precision of opinion classification. The Prefix Tree is selected to improve the overall efficiency. This study adopts data extracted from Plurk to explore sentiment analysis of Chinese texts, while making additional effort to observe change in execution time after adding the Prefix Tree.
Keywords: Sentiment Analysis, Opinion Mining, Prefix Tree.

1. 前言

隨著網際網路日漸普及在人們的生活，逐漸拉近人與人之間的距離，進而許多的互動、分享、與連結的相關應用也陸續地出現，例如維基百科(Wikipedia)、社群網站(Social Media)、部落格(Blog)等等。而在社群網站中以微網誌(Microblog)最為亮眼，每日有上百萬用戶在平台上發表各式各樣的論點，其中則以情感方面的闡述佔大部分，因此衍生出龐大的資料，使得關於文章的分析研究及探勘也就隨之產生，以Plurk的微網誌為例，Plurk設立於 2008 年 5 月 12 日，2009 年 8 月使用人數突破百萬人，直到 2014 年 2 月統計其用戶已達到 600 萬人，近幾個月不重複的訪客達 314 萬人次，而發表數量高達 4,500 萬次，雖然相較於其他會員數量更多的Twitter與Facebook裡Plurk發表文章數量不算最多，但本研究以中文情緒分析為主，故以中文資料蒐集的前提下，Plurk更適合用來當作數據來源。而在Plurk上所發表的文章，文字長度被限制在210個字元以內，並具有時間可追溯性，且發布文章皆有作者自身意見、理念和情感的自我展現。

在討論情緒分析常伴隨著自然語言處理(Natural Language Processing, NLP)，由於英文語句中會將每個字詞用空白隔開，而中文語句則是許多文字連接在一起，因此在詞彙的分割上有許多艱難的問題需要克服，故中文和英文在語言處理上是不相同的，一般在建立中文的語言模型中，通常會選用Trigram或Bigram作為模型的基礎，而許多情緒分析研究也常使用此兩種分析模型或結合兩種模型的方式來改善中文分類的準確率，因此可知訓練一個適合的模型對於分類來說相當重要。

本研究CCLM[1]組合模型分析前處理，主要透過中研院所開發的中文斷詞系統(Chinese Knowledge Information Processing Group, CKIP)，來對文章進行語句斷詞，例如:"這"、"間"、"餐廳"、"服務"、"好"、"，"、"東西"、"也"、"美味"，並且透過貝氏分類器的演算法來提高情緒分析的準確率。而在CCLM組合模型分類在面臨字詞數量增加時，會因出現新的字詞而重新計算字詞總數。故本研究結合Prefix Tree來提升字詞分析效果，藉由Prefix Tree共同關鍵字節點特性來改善情緒分析效能。

2. 文獻探討

2.1 情緒分析

情緒分析(Sentiment Analysis)[2]又稱作傾向性分析(Tendentious Analysis)[3]、意見挖掘(Opinion Mining)[4]，文字訊息進行分析、處理、歸納和推理，從中找出具價值性的資訊，最早由Turney在 2002 年提出以非監督式演算法[5]，應用層面廣泛，例如識別用戶產品的構種屬性評價，對於微網誌上用戶針對政治候選人的意見調查[6]，而有些公司則透過微網誌用戶來了解自家產品的各種評論進行改善，且還能透過此方式來了解競爭的產品資訊或行銷方式，從中學習來改善公司的缺點[7]。

情緒分析包含兩個部分，第一部份是意見抽取對文章所表達意見的文字，而Gamon提到根據文字的長短，可分成文件層級(Document Level)與語句層級(Sentence Level)，不同層級有著不同的處理方式

[8]，第二部分則是分析主觀意見，分別為正面、負面和中立，如表 1 所示。

表 1 情緒分析意見分類

正面情緒	模糊情緒(中立)	負面情緒
喜歡、快樂 喜悅、高興	驚訝、不錯	生氣、難過 厭惡、害怕

有些研究議題甚至會先將文章作主觀與客觀的分類加以提升分類的成效。Kim提到，對於意見的定義包含四個元素組成，分別是觀持有人(Holder)、評論對象(Target)、評論觀點(Type)與評論本文(Text) ，例如，"最近新聞報導地溝油事件，造成多少社會大眾身體健康受影響，這種不肖食品業者都該死"第一句話僅有陳述事件但可擷取出文章當中的主題是"地溝油事件"，在第二與第三句才有包含與評論相關的情感因素在內，因此若要藉由情緒分析在市場決策導向或公眾事務的選舉上作意見調查，不僅僅只是分析隱含在詞語中的情緒，更考量主題、發文者與陳述等條件，才能發揮情緒分析真正的效益。

2.2 最大事後機率

最大事後機率(Maximum a posteriori, MAP)[9]，可看作是貝氏定理的一種特定形式，是一個事件發生後，由點估計的方式來觀察難以統計機率數據，而此估計方式與最大似然率(Maximum Likelihood, ML)的差別在於最大事後機率有採用事前機率分佈的考量，舉例說明有 10 瓶水，其中 5 瓶水有糖份，5 瓶是純水，第一次喝掉的純水的機率 50%，第二次喝掉 45%，第三次喝掉 40%以此類推，因此想要連續喝到純水可透過ML在第一次喝跟第二次喝到純水的機率較高，但若增加可選擇喝水的種類，愛喝含糖份的人比例佔 80%，愛喝純水的人比例佔 20%，則須加入事前機率的判別，此時就要使用MAP來計算，可得知喝到含糖份的水比例較高，而MAP計算如下式(1):

$$\hat{\theta}_{MAP}(x) = {}^{argmax}_{\theta} f(x|\theta)g(\theta) \qquad (1)$$

假設目前要藉由觀察x來估計整體的θ參數，$g(\theta)$為事前機率，而$f(x|\theta)$則為事後機率，經由整體計算得出最大的值則為最佳解。

2.3 貝氏分類器

貝氏分類器是基於貝氏定理(Bayes' Theorem)透過概率統計與監督式學習的方式進而實現分類的方法，假設欲計算某篇文章d隸屬某一類別c，其以貝氏機率數學式 (2)所示:

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)} \qquad (2)$$

$P(c)$為事前機率(Prior Probability)，表示原先觀察分類c所分佈的概率，$P(c|d)$則為事後機率(Posteriori Probability)，表示在經過d文章中的特徵訓練後分佈在c分類中的概率，並以最大事後機率(Maximum-a-Posteriori)表示成如下式(3):

$$c_{map} = \underset{c \in C}{argmax} P(d|c)P(c) \qquad (3)$$

因要比較文章d在所有分類相似率的分母皆相同，故可將分母省略，並假設文章d中的特徵字詞分佈$(t_1, t_2 \ldots t_n)$皆相互獨立，如下式(4):

$$\begin{aligned} c_{map} &= \underset{c \in C}{argmax} P(t_1, t_2, \ldots., t_n|c)P(c) \\ &= \underset{c \in C}{argmax} P(t_1|c) * P(t_2|c) * \ldots P(t_n|c) * P(c) \\ &= \underset{c \in C}{argmax} P(c) \prod_{t=1}^{n} P(t_n|c) \end{aligned} \qquad (4)$$

因假設特徵樣本互相為獨立的情況，故可藉由已知的機率模型來推測未知資料所屬近似的類別，越相近則所得近似值越大。採用貝氏分類最大的好處在於當新特徵樣本加入後，僅需要在機率分佈上作微幅調整，故將此方法應用在分類上擁有簡單與高效率的特性。

2.4 N- gram

N-gram[10]屬於一種語言統計模型，建立模型前會先選定N要採用的是Unigram、Bigram與Trigram到N-gram的其中一種模型基礎，再將訓練資料集透過選定N-gram模型來拆出每篇文章中的組合文字並建立到語料庫當中，而各種語言在不同的N-gram模型皆有不同的成效，以中文來說，一個有意義的單詞(Term)通常由兩個或兩個以上文字組合而成，例如:今天天氣不還不錯，在Unigram語言模型會將語句拆成全部拆成單字，而在Bigram語言模型將單字兩兩組合，Trigram語言模型則單字以三個為一組，因此這些組合上可能常會出現無意義的單字，原因在於中文將所有文字串聯在一起，在斷詞上較難準確掌握，反觀英文把每個字詞透過空格來劃分，每個單字在重新組合後還是能保有原意，故在英文使用此模型成效會比中文的成效較佳。

在完成語料庫後建立而成的統計資料，並可藉由MLT來計算在某個文件序列的條件下所出現下一個文字的機率，如下式(5):

$$P(w_n|w_{n-N+1}^{n-1}) = \frac{C(w_{n-N+1}^{n-1}w_n)}{C(w_{n-N+1}^{n-1})} \qquad (5)$$

代表在文章中第n個字，C表示為某個字w所出現的機率，因此若N-gram為trigram，也就是N=3的時候，推算下一個文字w_n的機率可表示為式(6):

$$P(w_n|w_{n-1}w_{n-2}) = \frac{C(w_{n-2}^{n-1}w_n)}{C(w_{n-2}^{n-1})} \qquad (6)$$

2.5 Prefix Tree

Prefix Tree[11]也被稱為單詞搜尋樹。是一種很特別的樹狀資料結構如圖 1，Prefix Tree具有可快速的依照字詞插入、尋找、高效率特性，特別適合用於龐大的資料，由於龐大資料處理特性Prefix Tree常被使用在Data Mining、Data Streams，基於CCLM面臨的龐大字詞數量時效能會逐漸下降，因此藉由Prefix Tree來改善CCLM組合模型效能。

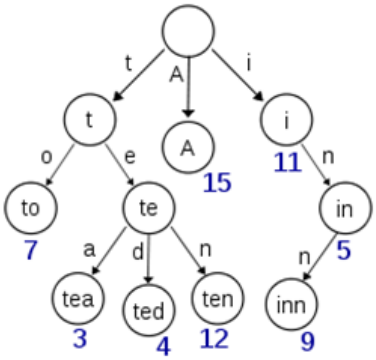

圖 1 Prefix Tree示意圖

3. 研究方法

本研究分為四大步驟如圖 2 所示，首先從Plurk蒐集資料，接著將蒐集的資料作前置處理，最後運用訓練模型並完成情感的分類器。

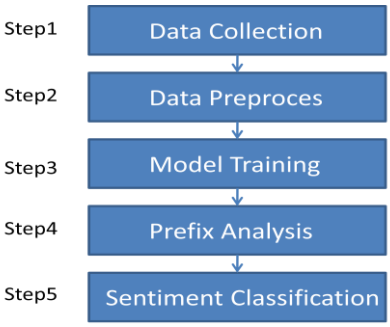

圖 2 研究方法處理步驟

3.1 Data Collection

本研究中訓練資料集(Dataset)主要的選擇來源為Plurk，因為相較於Twitter與Facebook，在Plurk上發表中文的使用者佔居多，因此能夠取得更多中文的文章資料。在Plurk上有開放API可供用以擷取相關文章，但在使用上仍有限制要求，故本研究透過獨立開發網路代理人(Network Agents)的方式來作為Plurk資料擷取的工具，透過代理人的方式便可繞過API存取限制在Plurk Search上擷取文章。在情感資料集進行訓練之前，需事先定義好新加入的文章之情感分類，而在本研究中將分類且明確定義正負面的情感，雖然可以藉由人工的方式來判斷文章的分類，再加入訓練資料集當中，但面對Plurk上數以萬計的文章，若以手動方式進行情緒分類會是一項相當耗費時間的工作，因此若能藉由文章中的情緒符號(Emoticons)如圖3所示來自動辨別文章的分類，便能省下相當多的時間。

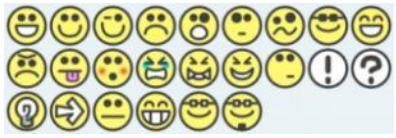

圖 3 Plurk情緒符號示意圖

面對Plurk上這麼多種的情緒符號，本研究從中挑選出使用率最高與正負面情緒最為明確的符號，分別是":-D"代表正面的情緒，而":-("則表示為負面的情緒，藉由這兩種情緒符號當作關鍵字，透過網路代理人在Plurk Search上蒐集從今年2014年至Plurk開放使用的2008年總共65,368篇文章，其中正面文章佔32,855篇而負面文章佔32,513篇，並且額外蒐集1,000筆的測試資料集透過人工標記的方式來判斷文章的分類，藉此可用以作為最後分類器測試準確率的依據。

3.2 Data Preprocess

雖然透過表情符號來判斷能夠有效的節省時間，但也相對地出現許多不必要的訊息，因此資料需要做前置的處理來移除非要的訊息，僅保留所需之資料。處理過程分為兩個步驟，第一步驟找出情緒符號所對應的語句，在一篇文章當中可能包含許多的情感符號，而每個情感符號所對應的語句不近相同，例如":-(天氣好冷，趕快收拾去洗熱水澡吧:-D"，因此"天氣好冷"的語句對應":-("的情緒，而"趕快收拾去洗熱水澡吧"的語句對應":-D"的情緒，所以要將每段語句透過對應情緒分切出來必須事先定義對應位置，故本研究將所蒐集的文章，依照情緒符號對應的語句分為4個案例來統計，其中分別為左、中、右與全部，如圖4所示。

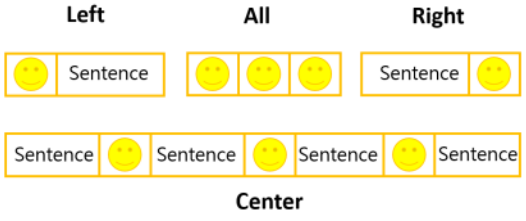

圖 4 Plurk文章中語句對應情緒位置

經統計後約有75,756筆文章裡發現到大多數的語句中情緒符號在句子右方出現有49,688筆文章比例為66%與出現在句子中間22,303筆文章，比例為29%兩者居多，故將過濾不屬於這兩類案例的文章，

並保留特徵屬於右方與中間的案例，而若位置處於中間且夾雜多個情緒符號的案例，則會透過每個情緒符號來分割並對應其左方的語句。

第二步驟將所有對應情緒的語句再進一步過濾，雖然當初蒐集的關鍵字有選用特定的正負面表情符號，但在蒐集的所有的文章仍會參雜許多不同的表情符號，所以需要再作第二次的過濾，去除掉非選用之表情符號，經過第二次過濾後從65,368篇文章轉變成52,694則語句並存放在訓練資料集當中。

3.3 Model Training

本研究在Model的建立，採用CKIP處理中文斷詞，將所有訓練資料集中的語句分拆成字詞，並標記原語句的情緒，如圖 5 所示。

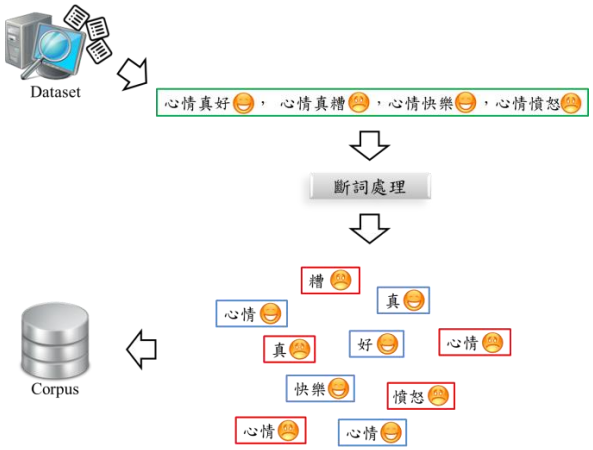

圖 5 透過CKIP處理資料集中文斷詞範例

斷詞後將所有的字詞存放到語料庫(Corpus)當中，並透過此語料庫再建立出兩個模型，分別是CKIP語言模型(CKIP Language Model, CLM)與CKIP組合式語言模型(CKIP Combined Language Model, CCLM)，例如: CKIP斷詞結果，心情(正情緒)、心情(負情緒)、真(正情緒)、真(負情緒)、好(正情緒)、糟(負情緒)、心情(正情緒)、心情(負情緒)、快樂 (正情緒)、憤怒(負情緒)，當中"心情"正負情緒比例為2/4，"真"正負情緒比例為 1/2，"好"正情緒為 1 負情緒為 0，以此方法將字詞匯總在CLM匯總語料庫中字詞與情緒出現的頻率中。

而在CLM當中觀察可發現在將語句拆成字詞後，有些原意會被改變，比如說"我不熱愛看書"，經過CKIP的處理後會拆成，"我"+"不"+"熱愛"+"看書"，因此不喜歡的原意就被改變了，故先前研究提出一種以CKIP斷詞與Bigram方式的CCLM，來改善CLM的不足，建立方式一樣先由語料庫中取出各字詞，並將這些字詞透過Bigram的形式作組合，如圖 6 完成所有字詞組合後，一樣匯總出經組合處理後語料庫中字詞與情緒出現的頻率，便可完成CCLM。

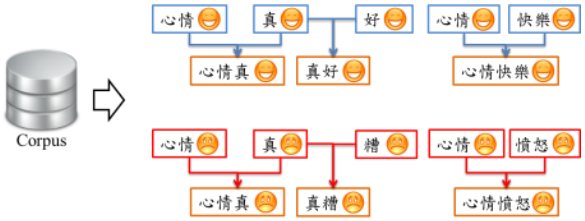

圖 6 CKIP結合Bigram組合字詞範例

3.4 Prefix Analysis

因CCLM模型內擁有Bigram形式組合的龐大字詞數量，導致CCLM模型運作時效能減低，故本研究利用Prefix Tree建立樹狀結構，將字詞及正負情緒機率從模組中取出如圖 7，並且記錄字詞情緒出現的頻率，透過Prefix Tree高效率查詢的特性加快搜尋字詞時間，達到更高效能的實驗結果。

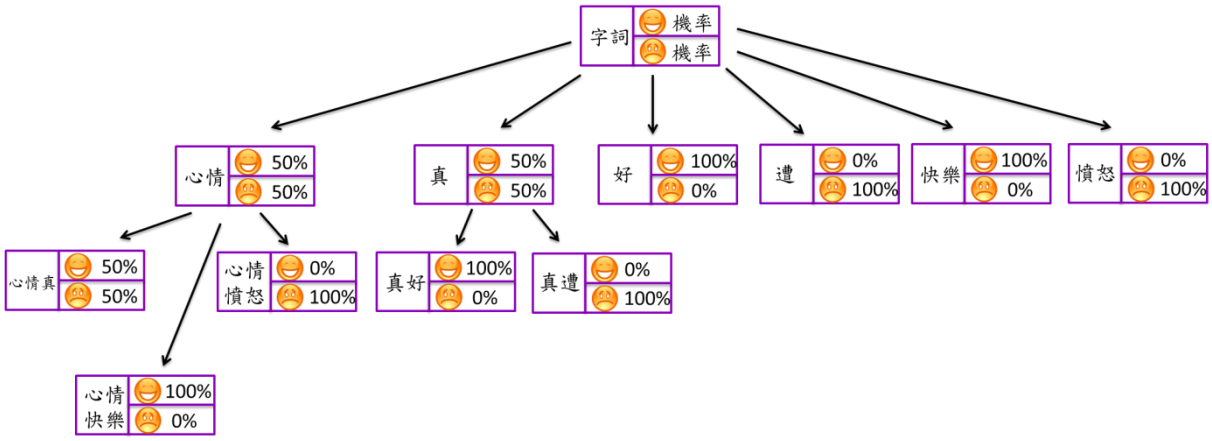

圖 7 Prefix Tree 範例圖

3.5 Sentiment Classification

本研究中採用貝氏分類器的方式搭配組合式語言模型來達成情緒分類，故也改善貝氏特徵計算方式。如式 (9):

$$C_{NB} = \underset{c_s}{argmax}\, P(c_s) * \prod_{t \in T} P(t|c_s) \quad (9)$$

假設目前有一筆文章d，在字詞特徵的擷取透過CKIP來處理斷詞之字詞特徵表示為Y集合，並將Y集合內的字詞以Bigram形式組合，產生組合式字詞特徵為X集合，Cs代表正面與負面情緒分類，在$P(x|c_s)$表示在Cs情緒分類條件下出現x的機率，若x>0，則去除含x字詞的y，若x<0 反之則保留y，最後將X∪Y並重新組合成T集合，以圖 8 為例。代入特徵字詞最後會計算出與正負面的C_{NB}值，最後會將文章d分類給C_{NB}值最大的類別。

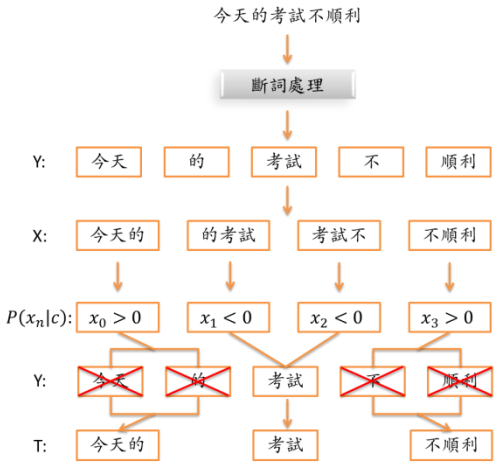

圖 8 文章分類字詞特徵前處理

4. 研究結果

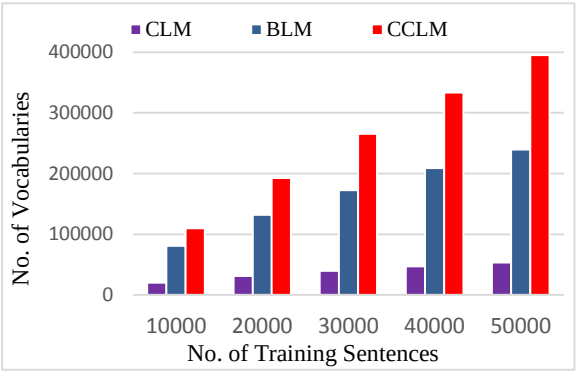

圖 9 BLM、CLM與CCLM字詞數統計圖

本研究在Plurk上蒐集1,000篇文章，並採用人工判斷情緒分類的方式來建立測試資料，並將訓練資料集中的52,694筆語句拆成為以每次增加10,000筆語句為一個測試單位的基礎下在作改善後之比較，藉此觀察在改善前後的訓練文章數量各種語言模型的字詞數、執行效率與準確率的統計結果之比較，另外除了CLM及CCLM兩組模型，也加入了Bigram語言模型(Bigram Language Model, BLM)的測試。以字詞數來看可從圖9觀察到，所有模型在不同訓練資料集數量下都呈現線性成長，CCLM字詞成長幅度為最大，BLM位居第二，只有CLM字詞數成長幅度最小，在50,000的訓練語句下CLM與CCLM兩者差距來到34萬筆字詞。

以準確率來看，在訓練資料集的語句數少的時候，採用BLM可以有相當不錯的準確率，但隨著訓練語句的增加，各種模型準確率卻是往下滑，據觀察在這中間因有不少的雜訊資料產生而影響準確率，但隨著資料增加到20,000則語句之後，CCLM比CLM與BLM的準確率皆有較佳的表現，雖然CLM準確率一度落後，直到資料增為50,000則語句後準確率才達到八成，但CLM與CCLM也已達到準確率的新高點。如圖10所示。

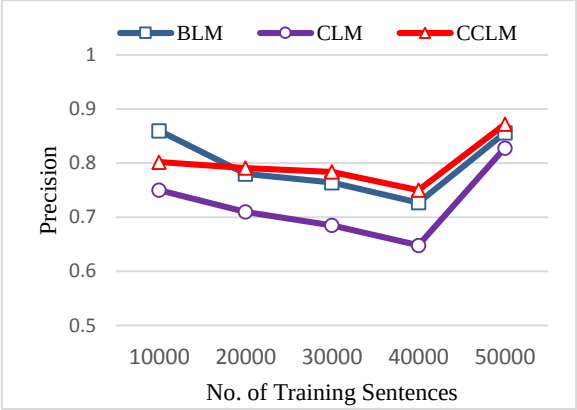

圖 10 BLM、CLM與CCLM準確率比較結果

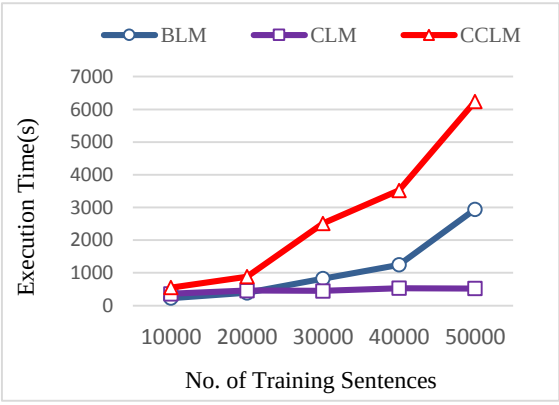

圖 11 BLM、CLM與CCLM字詞數統計圖

以圖11執行效率來看，CCLM執行計算時間最久，而其次為BLM與CLM，與字詞統計結果呈正比，由此可見當CCLM與BLM字詞數越多時，對整體執行的效率的影響越大，且假設訓練語句的數量持續增加，執行時間將會呈現指數的成長，而相對的

CLM在執行的速度上是最快的，且執行時間的成長也較不明顯。

由於CCLM字詞數持續增加的情況下，使得字詞統計結果影響整體的執行效率，因此建立Prefix Tree以及字詞索引來改善CCLM模型，從圖12執行效能來看，改善後的執行效能比原先的執行效能來的高。

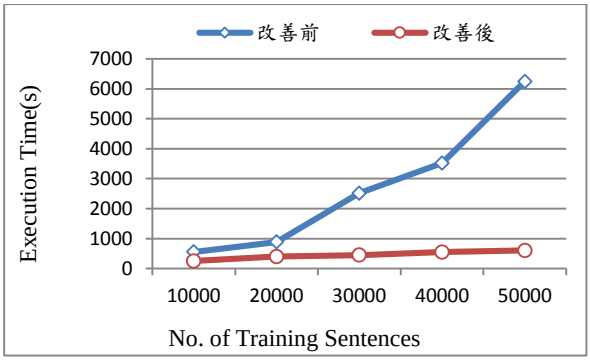

圖 12 改善後CCLM字詞數統計圖

5. 結論

本研究透過Prefix Tree來改善CCLM組合模組對於情緒分析效能不佳的情況，由於Prefix Tree每一個節點的字符都擁有相同的前綴，在搜尋時會減少無謂的字符串比較，故當樹狀結構持續成長時，也不會影響搜尋正負情緒機率的計算時間，本研究以人工標記的1,000篇文章來進行正負面情緒分類，綜合以上實驗結果可證明，在不影響準確率情況下，Prefix Tree擁有較高效率處理能力，且能節省計算時間，但相對地也佔用較多的記憶體空間。而隨著不同模組字詞數量的增加來訓練Prefix Tree字詞串，讓Prefix Tree分析字串效能提升，而在資料蒐集的部分，目前僅透過":-D"與":-("兩種特徵情緒符號來達到訓練文章的分類，而為求減少雜訊產生，因此過濾掉所有不含在這兩種情緒符號的所有語句，而在未來的研究上將近一步加強所有情緒符號的識別，且希望將此系統研究成果應用於實際的銷售市場或排名的預測研究。

參考文獻

[1] Y.Q. Chen, "Using Sentiment Analysis to Predict Public Behaviors," Master dissertation, Shu-Te University. Taiwan, 2014.

[2] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment Classification using Machine Learning Techniques," Empirical methods in natural language processing, Vol 10, pp. 79-86, 2002.

[3] M. Zhang, Y. Peng, Y. Fan, D. Li, X. Lin, and X. Wu, "Research on Chinese Orientation Analysis," The Chinese propensityanalysis evaluation seminar, pp. 38-45, 2008.

[4] A. Esuli, and S. Fabrizio, "Sentiwordnet:A publicly available lexical resource for opinion mining," Proc. of the 5th Conference on Language Resources and Evaluation (LREC'06), Vol. 6, pp. 417–422, 2006.

[5] X.L. Li, J.M. Lin, and Z.Z. Shi, "A Chinese Web Page Classifier Based on Support Vector Machine and Unsupervised Clustering," Chinese Journal Of Computers-Chinese Edition, Vol. 24, No. 1, pp. 62-68, 2001.

[6] A. Go, B. Richa, and H. Lei, "Twitter Sentiment Classification using Distant Supervision," CS224N Project Report, pp. 1-12, 2009.

[7] P.D. Turney, "Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews," Proc. of 40th Annual Meeting on Association for Computational Linguistics, pp. 417-424, 2002.

[8] M. Toews, D.L. Collins, and T. Arbel, "Maximum a Posteriori Local Histogram Estimation for Image Registration," Medical Image Computing and Computer-Assisted Intervention-MICCAL, pp. 163-170, 2005.

[9] A. Park, and P. Paroubek, "Twitter as a Corpus for Sentiment Analysis and Opinion Mining," Proc. of the 7th International Conference on Language Resources and Evaluation, pp. 42-44, 2010.

[10] P.F. Brown, and P.V. Desouza, "Class-based n-gram models of natural language. Computational linguistics," Computational Linguistics, Vol. 18, No. 4, pp. 467-479, 1990.

[11] G.M. Liu, H.J. Lu, W.W. Lou, B.X. Ya, and X.Y. Jeffrey, "Efficient mining of frequent patterns using ascending frequency ordered prefix-tree," Data Mining and Knowledge Discovery, pp. 249-274, 2004.