網路社群回應評分系統：以 Facebook 為例

楊燕珠
大同大學資訊經營研究所
yjyang@ttu.edu.tw

沈佳弘
大同大學資訊經營研究所
79912020@ttu.edu.tw

摘要

網路社群應用已日漸普及，接觸的媒介也趨於多元，人們現在可以非常容易地瀏覽與回應網路資訊（包括圖、文、影音等形式）。然而，其中有很大一部分的文字，尤其是微網誌的回應，除發言者之外，幾乎不具意義；在一個每日數百則以上回應的社群單位中，類似：問候、廣告、喃喃自語、與主題無關的爭執等，經常佔了一半以上，人們無法在短時間中抓到數則相關而有意義的回應，該主題可能就無法持續被正常地關注討論。本研究提出一套評分系統，為 Facebook 中塗鴉牆上的回應評分，讓使用者可以直接瀏覽評分較高的回應，提高後續回應品質。

研究中透過 (一) 語意分析 (二) Meta 特徵比對 (三) 文本相似度計算等三個步驟，將所有回應逐次從上一個程序較相關的部分再細分為相關和不相關兩群，直到所有程序執行完畢。根據本研究的實驗結果，在誤判率 FPR(False Positive Rate)為 0.1 時，就可以得到最高 0.9 的 TPR(True Positive Rate)。

關鍵字：社群、評分、Lesk Algorithm、Kappa、Facebook

壹、 前言

幾年前我們還使用個人部落格(blog)抒發意見，到企業官網留言給客服信箱，用即時通訊軟體和朋友聊天；如今這些事情全都集中在社群媒體中實現，不論是美國總統 Obama，或是流行歌手 Lady Gaga，都和七億多個 Facebook 使用者一樣，在這個當今最大的社群媒體上分享觀點，交流資訊。以 Obama 總統的粉絲專頁為例，幾乎每天會有一則以上的發言，每則發言會有數千至數萬不等的回應，雖然大部分回應與 Obama 的發言極度不相關(包括問候、廣告和無關主題的謾罵)，但還是有少部分是相關且較有意義的；然而 Facebook 介面並沒有回應的重要性排名機制，甚至沒有搜尋留言的功能，因此使用者難以快速瀏覽全部留言，這和前述大部分回應與 Obama 發言極度不相關可能有相當程度互為因果的關係。

本研究提出一個流程，經過語意分析和特徵萃取，取得資料進行相似度計算。我們將在第貳章回顧過去文獻與相關研究，第參章會介紹這個流程的設計，並且介紹語意分析、特徵萃取以及相似度計算的演算法。第肆章有針對這個流程設計的實驗設計說明，以及實驗結果和分析，相關討論和本研究結論則整理在第伍章。

貳、 相關研究

一、 語意分析與銷歧

語意分析與銷歧是自然語言文本處理很重要且經常是第一個需要處理的問題。為了後續計算文本間的相似度，對於一字多義的字詞，必須進行語意銷歧(Word Sense Disambiguation)；以英文為例，「catch a cold」和「it's cold」的「cold」，涵義(sense)就完全不同。

語意銷歧的方法大致可分為非啟發式和啟發式兩大類。非啟發式透過文件集本身各文本不斷地相互比較，逐漸將語意分群，這種方式所需的硬體支援及時間會隨著文件集的擴大而呈指數成長。啟發式又分為(一) 鄰近字或句型式和 (二) 字典式；第一種透過訓練資料，得知一個字鄰近某些特定字時，可能具備的意義，這種方式較適合特定領域的文章或討論，他們有經常出現的詞彙和句型；字典式則是利用字典檔中，對字詞註解的文字，找出其同義字詞(synonym)、上位字詞(hypernym)、下位字詞(hyponym)等相關字詞，進而計算正確語意機率。

字典式中的經典作法之一，是 Lesk, Michael. (1986)提出 Lesk Algorithm，它是用詞彙註解中的最長匹配段落覆蓋值(max-overlap)，來解析出字詞的上位字詞、下位字詞等，從而求得正確語意。例如：

PINE 有兩個詞義：

(1) kinds of **evergreen tree** with needle-shaped leaves

(2) waste away through sorrow or illness

CONE 有三個詞義：

(1) solid body which narrows to a point

(2) something of this shape whether solid or hollow

(3) fruit of certain **evergreen tree**s

由於 PINE 的第 1 個詞義中和 CONE 的第 3 個詞義中 **evergreen tree** 重複，max-overlap=2，標示為：

Pine#1 ⋂ Cone#3 = 2

```
function SIMPLIFIED LESK(word,sentence) returns best sense of word
      best-sense <- most frequent sense for word
      max-overlap <- 0
      context <- set of words in sentence
      for each sense in senses of word do
      signature <- set of words in the gloss and examples of sense
      overlap <- COMPUTEOVERLAP (signature,context)
      if overlap > max-overlap then
      max-overlap <- overlap
      best-sense <- sense
end return (best-sense)
```

圖 2-1：Simplified Lesk Algorithm (Kilgarriff and Rosenzweig, 2000) 的 SUDO code。

二、 Stemming

為了減少英文字詞中各種變化型降低相似度計算的準確度的情形，文本集在檢索或相似度比較前，會執行同義異型詞的統一，我們稱之為 stemming。處理英文文本集 stemming 較廣泛被利用和接受的作法，是 Martin F. Porter(1980)提出的 Porter stemming algorithm，他以非常簡潔的幾個原則，例如：SSES 字尾替換為 SS，IES 字尾替換為 I，SS 字尾替換為 SS，S 字尾直接刪除等等，將英文單字帶入相對應的轉換格式，還原出字根。

三、 單字權重

每個單字在文章中的重要性都不一樣；一般來說，在一個文本中，出現頻率較高的字詞，通常和這個文本的相關性較高；而從整個文本集的角度來說，若某個字詞在各個單一文本都出現的頻率較高，重要性和相關性反而應該是較低的；基於這個考量，在資料檢索領域中，通常會採用 tf-idf 公式來平衡權重，tf 是詞頻，字詞 t 在 d 文件的頻率表為 tf(t,d) ；idf 是反轉文件頻率，其公式如下：

$$\text{idf}(t) = \log \frac{D}{\{d:t \in d\}} \qquad \text{(公式 2-1)}$$

其中 D 是文件集中的文本數，{d:t∈d} 是包含字詞 t 的文本數。字詞 t 在文件 d 中的 tfidf 值表為：

$$\text{tfidf}(t,d) = \text{tf}(t,d) * \text{idf}(t) \qquad \text{(公式 2-2)}$$

四、 相似度計算

Jaccard 係數(Jaccard Paul, 1901)是衡量兩筆資料相似度時，最廣泛被使用的衡量標準，假設有 A,B 兩個文本，X 是 A 文本所有單字的集合，Y 是 B 文本所有單字的集合，則 A,B 的 Jaccard 係數為：

$$\text{Jaccard}(A,B) = \frac{X \cap Y}{X \cup Y} \qquad \text{(公式 2-3)}$$

其中 X∩Y 是 A,B 兩個文本的重複字數，X∪Y 是 A,B 兩個文本的總字數減掉兩個文本的重複字數。

五、 效能評估

(一)專家評分一致性評估

為了瞭解多位專家對於同一件事物的看法是否一致，乃至於一份問卷回收後是否具可信度，我們必須進行一致性評估。Cohen, Jacob (1960)提出 Kappa 係數 κ，作為人際信度(inter-rater agreement)的衡量方法，稱為 Cohen's Kappa(記作 κ)，κ 的公式如下：

$$\kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)} \qquad \text{(公式 2-4)}$$

其中 Pr(a)是評分者間觀察的一致性，Pr(e)是隨機下期望的一致性，意即評分者獨立情況下，隨機假設時一致性的期望值。

可惜 Cohen's Kappa 只能對評分者數量為 2 時進行計算，於是 Joseph L. Fleiss(1971)提出可以針對多位評分者適用的 Kappa 係數，稱為 Fleiss' Kappa。Fleiss' Kappa 的演算法，首先假設 N 是問卷題數；n 評分者人數；k 是每一題有幾個選項；問卷題目編號 i 由 1 到 N 號；選項編號 j 從 1 到 k；n_{ij} 代表第 i 題選擇第 j 個選項的評分者人數，如此答 j 的比例 P_j 為：

$$p_j = \frac{1}{Nn}\sum_{i=1}^{N} n_{ij} \qquad \text{（公式 2-5）}$$

而所有評分者對第 i 題的兩兩相比的一致度為：

$$P_i = \frac{1}{n(n-1)}[(\sum_{j=1}^{k} n^2{}_{ij}) - n] \qquad \text{（公式 2-6）}$$

此時可以計算 Cohen's Kappa 所需的各變數 Pr(a)和 Pr(e)：

$$\Pr(a) = \frac{1}{N}\sum_{i=1}^{N} P_i \qquad \text{（公式 2-7）}$$

$$\Pr(e) = \sum_{j=1}^{k} p_j^2 \qquad \text{（公式 2-8）}$$

將 Pr(a)和 Pr(e)帶入公式(2-1)便可得出 Kappa 值。

針對 Kappa 值高低所代表的涵義，Landis, J.R.; & Koch, G.G. (1977) 提出一個詮釋的標準，如表 2-1。

表 2-1：Kappa 值高低所代表的涵義。

κ	Interpretation
< 0	Poor agreement (差)
0.01 – 0.20	Slight agreement (輕微)
0.21 – 0.40	Fair agreement (普通)
0.41 – 0.60	Moderate agreement (中等)
0.61 – 0.80	Substantial agreement (高)
0.81 – 1.00	Almost perfect agreement (近乎完美)

(二)實驗效能評估

文件分類的效能評估，通常先將結果整理成文件分佈表，如表 2-2。True 表示「正確(或答對）」的類別，false 表示被錯分；而 positive 表示「分類為相關」，negative 表示「分類為不相關」。我們用這四個數值來計算「正確率」(accuracy)、「 精確率」(precision）和「召回率」(recall)。

表 2-2：文件分佈表

A (true positive)	B (false negative)
C (false positive)	D (true negative)

$$\text{正確率（accuracy）}=\frac{A+D}{A+B+C+D} \qquad \text{（公式 2-9）}$$

$$\text{精確率（precision）}=\frac{A}{A+C} \qquad \text{（公式 2-10）}$$

$$\text{召回率（recall）}=\frac{A}{A+B} \qquad \text{（公式 2-11）}$$

由於不相關(false)的文件數經常遠大於相關(true)的文件數，且通常我們只關心少部分分類為相關的文件，因此 D (false negative)通常遠大於 A、B 或 C，正確率在這種情況下就會趨近於 1，參考價值很低。而精確率和召回率之間，常有連動的影響，因此為了兼顧這兩個數據，避免出現高精確率加低召回率，或低精確率加高召回率的結果，我們會採用精確率和召回率的加權平均，所謂的 F-measure，其公式為：

$$F_\beta = (1+\beta^2)\cdot\frac{\text{precision}\cdot\text{recall}}{(\beta^2\cdot\text{precision})+\text{recall}} \qquad \text{（公式 2-12）}$$

當 β=1 的時候，F-measure 就等於精確率和召回率的調和平均，意即：

$$F_1 = 2\cdot\frac{\text{precision}\cdot\text{recall}}{\text{precision}+\text{recall}} \qquad \text{（公式 2-13）}$$

從精確率、召回率到 F-measure，雖然可以表現出整體的效能，但還沒有加上排序的表現；實際檢索的應用上，順序是很重要的，也就是說，我們瀏覽檢索結果，通常只看前一兩頁，可能是前 10 則、20 則，後面其實不重要；要將排序的表現加入效能評估來觀察的話，可以採用 11-point Interpolated Average Precision 法，11-point Interpolated Average Precision 是將召回率分成 10 等分，產生 0.0, 0.1, 0.2, ……, 1.0 等 11 個點，這些點的值代表在這個召回率和上一個點的召回率間的曾出現的最高精確率。

召回率也有人稱為敏感性(sensitivity)或是 TPR(True Positive Rate)，因為它是相關且被分類為相關的文本數占全部相關文本數的比例；相對於敏感性，不相關且被分類為不相關的文本數占全部不相關文本數的比例，稱為特異性(specificity) ；那麼(1-特異性)就是誤判為相關的比例，可表示為 FPR(False Positive Rate)。若我們以敏感性為縱座標，以 FPR 為橫坐標，

就可以畫出一條 ROC (Receiver Operating Characteristic)曲線。

當敏感性固定時，FPR 越小，代表誤判較少就能達到同樣的召回率，這是一個效能較好的系統。因此可以說，當一個 ROC 曲線越偏左上，這個系統效能就越好。

參、 研究方法

一、研究架構與流程設計

本研究主要透過對文本的語義分析，計算文本間的相似度(關聯性)，加上 meta-data 解析達到評分的目的。流程設計如圖 3-1。

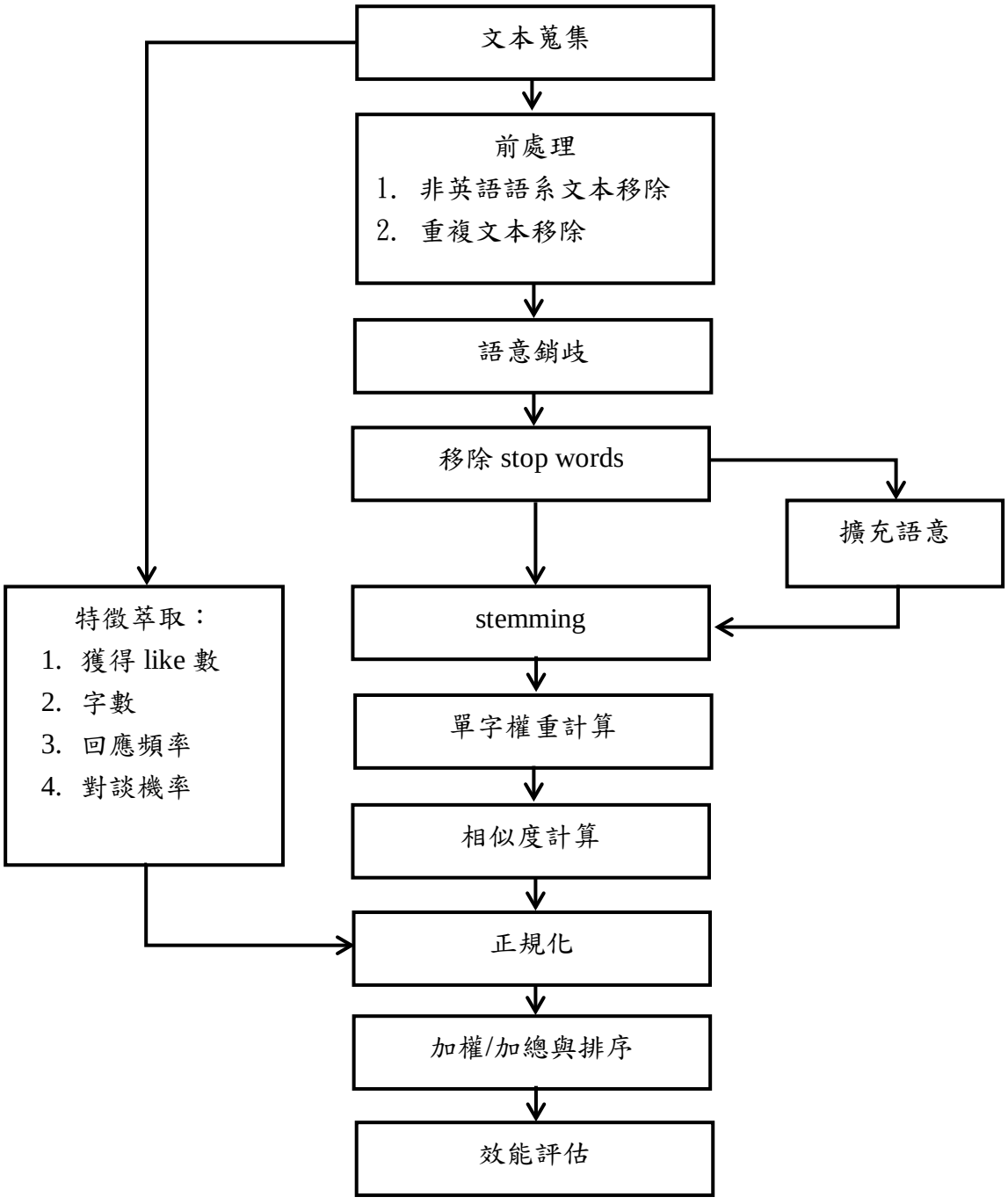

圖 3-1 流程設計

二、 文本蒐集

本研究以美國總統 Obama、藝人 Lady Gaga 和 Facebook 公司的 Facebook 粉絲專頁為取樣目標，蒐集這三個目標的塗鴉牆(wall)為文本集，蒐集範圍為 2011 年 8 月 1 日 00:00 到 8 月 31 日 23:59 為止，由粉絲專頁擁有者(以下簡稱「版主」)發言的討論串。

三、 前處理

(一) 排除非英語系文本：

本研究的後續步驟均以英語為處理背景，因此先排除非英語系文本。排除方式是以 WordNet 檢索每一則回應前 10 個超過 2 個字元的單字，若超過 5 個單字查非英語，則將該回應從文本集剔除。

(二) 排除重複文本：

本研究所使用的所有回應，經統計後，每個討論串中約有 5%的同一使用者的重複回應，通常是廣告、重複點擊送出鈕或是惡意重複留言造成。因此我們在一開始就將所有回應逐一與同一使用者時間較晚的回應比對，並刪除將所有相似度 90%以上的較晚回應，僅留存時間最早的一則。

四、 語意銷歧

由於本研究的文本集相當龐大，涉及議題也非特定領域，因此字典式語意銷歧應是最適合的實作方式，我們採用 Lesk Algorithm 來實作字典式語意銷歧。

Lesk Algorithm 步驟有二：

(一) 詞意列表

詞意列表需要有一個足夠規模和信度詞典，本研究所使用的文本集為英文文本；在英文語系中，WordNet 是常被使用的語意資料庫，它收錄了大量的英文詞彙，並且提供多種檢索結果的條列格式，易於以程式解析英文詞義。

本研究採用的 WordNet 3.0 版，收錄了 155287 個獨立字詞，內含 117,798 個名詞，11,529 個動詞，21,479 個形容詞和 4,481 個副詞。

WordNet 會將文本片段(Parts of Speech ,POS)中的名詞、動詞、形容詞和副詞用一組同義字和註解表達出來，有時會加上以雙引號括起的例句，例如 car 這個詞的第一行為：

car#1, auto#1, automobile#1, machine#6, motorcar#1 (a motor vehicle with four wheels; usually propelled by an internal combustion engine) "he needs a car to get to work"。

這表示 car 的第一個詞義和 auto 的第一個詞義，automobile 的第一個詞義，machine 的第六個詞義和 motorcar 的第一個詞義，都同為「a motor vehicle with four wheels; usually propelled by an internal combustion engine」的意思，he needs a car to get to work 則為例句。

(二) 計算正確語意

藉由將每個前處理後的文本的每個字的詞意列表，我們可以進行 Lesk Algorithm 的主要部分，也就是找出其註釋句子中的 overlap 字詞數，以最長匹配數(max-overlap)所屬的詞意為正確詞意。

五、 移除 stop words

文章中有許多經常出現的字詞，通常並不影響整個句子的相似度比對，例如英文中的 a、an、as……等等，將這些字詞移出會大幅減低待處理文件的大小，通常可以減少 30% ~ 50%。本研究所使用的 WordNet 3.0 版 stop words 共有 199 個。

六、 擴充語意

經過語意銷歧和移除 stop words 之後，我們可以為所有文本加上擴充語意，擴大相似度計算的容許度，以增加相似度計算的準確度。WordNet 除了會提供同義字詞，還可以透過參數指令列出相關詞彙組，對於名詞會提供上位字詞、下位字詞、涵蓋字詞(Holonyms)；動詞會提供上位字詞、下位字詞、含意字詞(Verb Entailment)，這些都可以讓我們在執行 Lesk Algorithm 時產生的詞意查詢結果後接續取得，當成該字詞擴充語意的語料。

七、 Stemming

為提高相似度計算的正確率，我們將文本進行一次 stemming，本研究使用最廣為被接受的 Porter Stemming Algorithm 進行 stemming。

八、 tf-idf

我們用 tf-idf 來計算每個文本中各個字詞的重要性，tf 是字詞 t 在版主發言或是一則回應中出現的頻率；公式中的 D，也就是文本集總數，我們用的是該目標粉絲頁的所有回應數加上版主發言數，而非三個目標的所有回應數加上版主發言數，這是考量三個目標的討論領域有所不同，文本集以一個目標為單位較為合理。

九、 相似度計算

經過以上各個步驟地整理，至此我們可以開始計算相似度，本研究採用 Jaccard 係數公式，計算版主發言和每一則回應的相似度；Jaccard 係數將兩個文本的重複字數，除以兩個文本的總字數減掉兩個文本的重複字數；但考量每個字詞的重要性不同，因此我們將公式中的字數改為相對應的 tfidf 值總和，公式如：

$$\text{Jaccard}(A,B) = \frac{\sum tfidf(X \cap Y)}{\sum tfidf(X \cup Y)}$$

$$= \frac{\sum_1^p tfidf(t_p, d_A)}{\sum_1^m tfidf(t_m, d_A) + \sum_1^n tfidf(t_n, d_B) - \sum_1^p tfidf(t_p, d_A)} \quad \text{(公式 3-1)}$$

其中 A 為版主發言文本，B 為一則回應文本；X 是 A 文本所有單字的集合，共有 m 個元素；Y 是 B 文本所有單字的集合，共有 n 個元素；X∩Y 的元素共 p 個。

十、 特徵萃取與正規化

除了回應文本和發言文本的相似度計算之外，還有許多 meta-data 可以作為一個回應的特徵，這些特徵都可以幫助我們辨識一個回應是否重要，我們採用其中 4 個 meta-data 為本研究的特徵。

(一)獲得 like 數

Facebook 的每一則回應下方都有一個 Like 按鈕，通常瀏覽者若認同該則回應，便會按下 Like 按鈕；一則回應有一個以上的瀏覽者按下 Like 按鈕之後，按鈕右側便會出現按下 like 按鈕的人總數。

我們將這個值正規化到 0 到 1 之間，公式如下：

$$\text{Like}=\frac{Like_{Get}}{Like_{Max}} \qquad \text{（公式 3-2）}$$

其中$Like_{Get}$是這則回應獲得的 Like 數，而$Like_{Max}$是這個討論串中最高 Like 數。

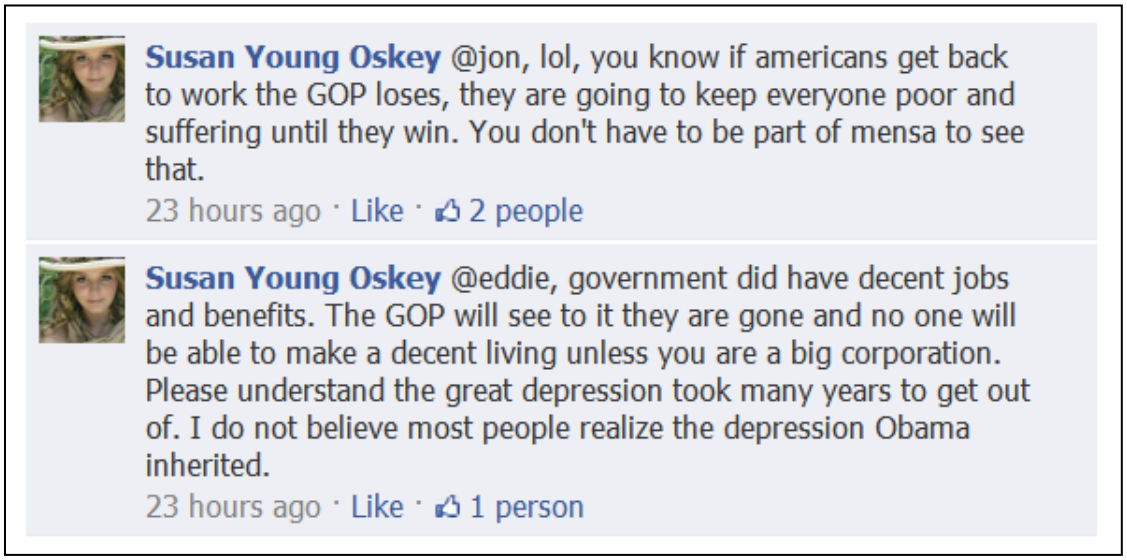

圖 3-3：Facebook 的回應

(二)字數

回應的字數對其重要性當然有影響，但 1000 個字的回應未必是 10 個字回應的 100 倍，為避免極端值影響，本研究統計每則回應的字數，以四分位距轉換成為 0.25 , 0.5 , 0.75 和 1 四種值，以 Q1 表最小四分位數，Q2 表中位數，Q3 表最大四分位數，則轉換公式如表 3-1 所示：

表 3-1 回應字數影響值轉換表

轉換前分佈位置	轉換後影響值
0<字數≦Q1	0.25
Q1<字數≦Q2	0.5
Q2<字數≦Q3	0.75
Q3<字數	1

(三)回應頻率

回應者的回應頻率表示其對此討論串的熱衷程度，但同樣地，為避免極端值影響，本研究以四分位距將回應頻率轉換成為 0.25 , 0.5 , 0.75 和 1 四種值，以 Q1 表最小四分位數，Q2 表中位數，Q3 表最大四分位數，則轉換公式如表 3-2 所示：

表 3-2 回應頻率影響值轉換表

轉換前分佈位置	轉換後影響值
0<頻率≦Q1	0.25
Q1<頻率≦Q2	0.5
Q2<頻率≦Q3	0.75
Q3<頻率	1

(四)對談機率

在網路討論區或是類似開放討論留言的互動式網頁，包括 Facebook，使用者經常會在留言開端寫下希望回應的對象的網路暱稱，例如圖 2 中的 Susan 發表了兩則回應，第一則回應 jon，第二則回應 eddie；藉由這樣的使用方式，我們可以找出那些人在這個討論串中可能在進行對話。我們將可能進行對談的回應賦予 1 的影響值，無進行對談的回應則賦予 0 的影響值。

肆、 實驗與討論

一、 實驗文本集內容

本研究相關實驗以美國總統 Obama、藝人 Lady Gaga 和 Facebook 公司的粉絲專頁，2011 年 8 月 1 日 00:00 到 8 月 31 日 23:59 為止的塗鴉牆文字為文本集，蒐集內容資訊列表如表 4-1。

表 4-1 實驗文本集內容——目標粉絲專頁內容資訊列表

粉絲專頁名稱	討論串數	回應總數	網址
Barack Obama	59	246,382	www.facebook.com/barackobama
Lady Gaga	30	235,368	www.facebook.com/ladygaga
Facebook	11	133,645	www.facebook.com/facebook

二、 評估系統

本研究以 Fleiss 的 Kappa 一致性係數來評估實驗結果。

我們請三位專家將每一個討論串的所有回應進行分類，每個回應都分到相關或不相關，再將三位專家回饋的結果與進行 Kappa 值一致性分析，以 Kappa 值高於 0.8 的討論串中，三位專家均認為相關的回應為正確答案，用本研究的評分系統計算 11-point Interpolated Average Precision，以觀察其評分效能。

三、 實驗設計與結果

(一) 實驗一：標準效能評估

本研究中每個回應經處理後都有五種屬性：(1)文本相似度(2) 獲得 like 數(3)字數(4)回應頻率(5)對談機率，在實驗一中，我們將這五個屬性依照第參章中所述各種方式正規化為 0 到 1 之間的值，然後加總後由大至小排序，實驗結果，三個目標的召回率、精確率、F1 Score 和 11-point Interpolated Average Precision 如表 4-2。

表 4-2 實驗一結果(R：Recall；P：Precision)

Barack Obama					Lady Gaga					Facebook				
Raw data		F1	Interpolated		Raw data		F1	Interpolated		Raw data		F1	Interpolated	
R	P		R	P	R	P		R	P	R	P		R	P
0.45	0.74	0.56	0.00	1.00	0.15	0.15	0.15	0.00	1.00	0.30	0.89	0.44	0.00	1.00
0.75	0.62	0.68	0.10	1.00	0.44	0.21	0.29	0.10	1.00	0.62	0.94	0.75	0.10	1.00
0.79	0.43	0.56	0.20	1.00	0.74	0.24	0.36	0.20	0.38	0.89	0.89	0.89	0.20	0.85
0.82	0.34	0.48	0.30	0.88	0.80	0.19	0.31	0.30	0.16	0.90	0.68	0.78	0.30	0.89
0.84	0.28	0.41	0.40	0.88	0.84	0.16	0.27	0.40	0.20	0.92	0.55	0.69	0.40	0.92
0.91	0.25	0.39	0.50	0.75	0.87	0.14	0.24	0.50	0.24	0.95	0.48	0.64	0.50	0.93
0.96	0.22	0.36	0.60	0.75	0.91	0.13	0.22	0.60	0.27	0.98	0.42	0.59	0.60	0.94
0.98	0.20	0.33	0.70	0.73	0.95	0.12	0.21	0.70	0.28	0.99	0.38	0.54	0.70	0.95
0.99	0.18	0.31	0.80	0.69	0.98	0.11	0.19	0.80	0.26	1.00	0.34	0.50	0.80	0.95
0.99	0.16	0.28	0.90	0.39	1.00	0.10	0.18	0.90	0.20	1.00	0.30	0.46	0.90	0.96
1.00	0.00	0.00	1.00	0.00	1.00	0.00	0.00	1.00	0.00	1.00	0.00	0.00	1.00	0.00

我們分別以 Raw data 和 Interpolated data 繪製 P-R 圖如圖 4-1。從圖表上可以看出，以內插法算出的 11-point Interpolated Average Precision 由於考量到之前平均的表現，精確率下降的趨勢較緩，經調整後，三個目標在召回率 0.1 的時候，都能夠達到 1.00 的精確率，顯然本系統在運作初期已正確判斷部分相關文本，我們接著畫出三個目標的 ROC 曲線，來比較本系統對三個目標的效能。

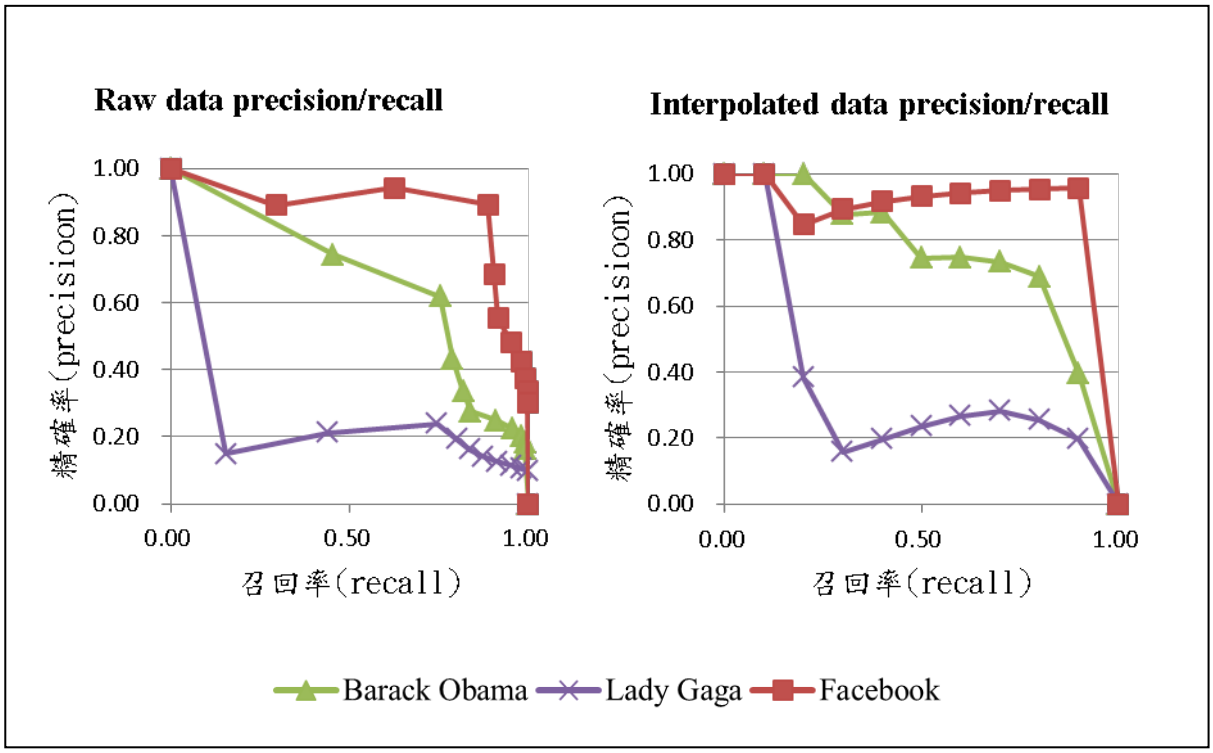

圖 4-1 以 Raw data 和 Interpolated data 繪製實驗一的 P-R 圖

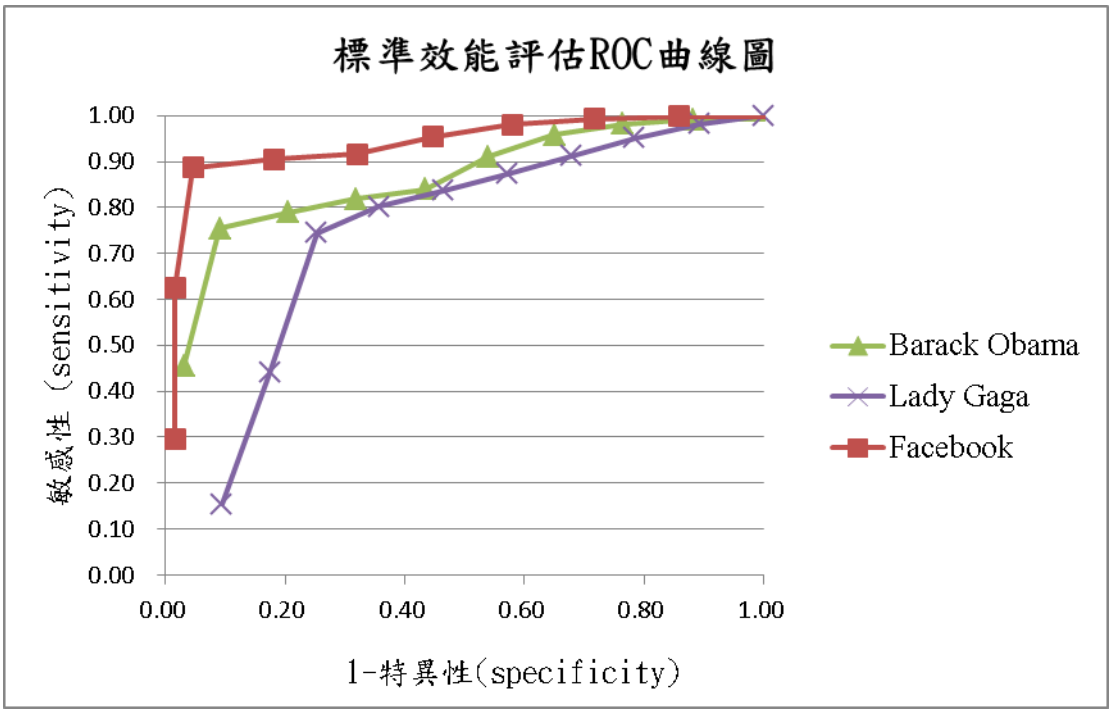

圖 4-2 ROC 曲線圖

從 ROC 曲線圖可以看出，最靠左上角的曲線是 Facebook，然後依次為 Barack Obama 和 Lady Gaga，表示本系統分辨 Facebook 粉絲專業的網友回應語版主發言相關性的效能最好。我們從圖 4-1(P-R 圖)可以看出 Lady Gaga 的精確率在召回率 0.2 的時候，就迅速滑落至 0.38 了；而在 ROC 曲線圖中，要達到同樣 0.8 的召回率，Facebook 只會產生 5%的誤判，Lady Gaga 卻必須伴隨將近 40%的誤判。

(二) 實驗二：加權效能評估

實驗二將每個回應的五種屬性影響權重個別降低到 10%，再與標準評估的效能做比較，加權公式如下：

$$\text{Rank}_W = W_{sim} \cdot V_{sim} + W_{like} \cdot N_{like} + W_{wc} \cdot N_{wc} + W_{pf} \cdot N_{pf} + W_{reply} \cdot V_{reply} \quad \text{（公式 4-1）}$$

其中V_{sim}是相似度，N_{like}是正規化後的 like 數，N_{wc}是正規化後的回應字數，N_{pf}是正規化後的回應頻率，V_{reply}是對談機率值，W_{sim}、W_{like}、W_{wc}、W_{pf}和W_{reply}是各項權重，我們對每個目標都依次將設為 0.1，各取得五組數據。

經過加權後的效能，我們以 ROC 曲線圖來做比較，圖 4-3 為 Barack Obama 粉絲專頁回應加權效能評估 ROC 曲線圖，圖 4-4 為 Lady Gaga 粉絲專頁回應加權效能評估 ROC 曲線圖，圖 4-5 為 Facebook 粉絲專頁回應加權效能評估 ROC 曲線圖。

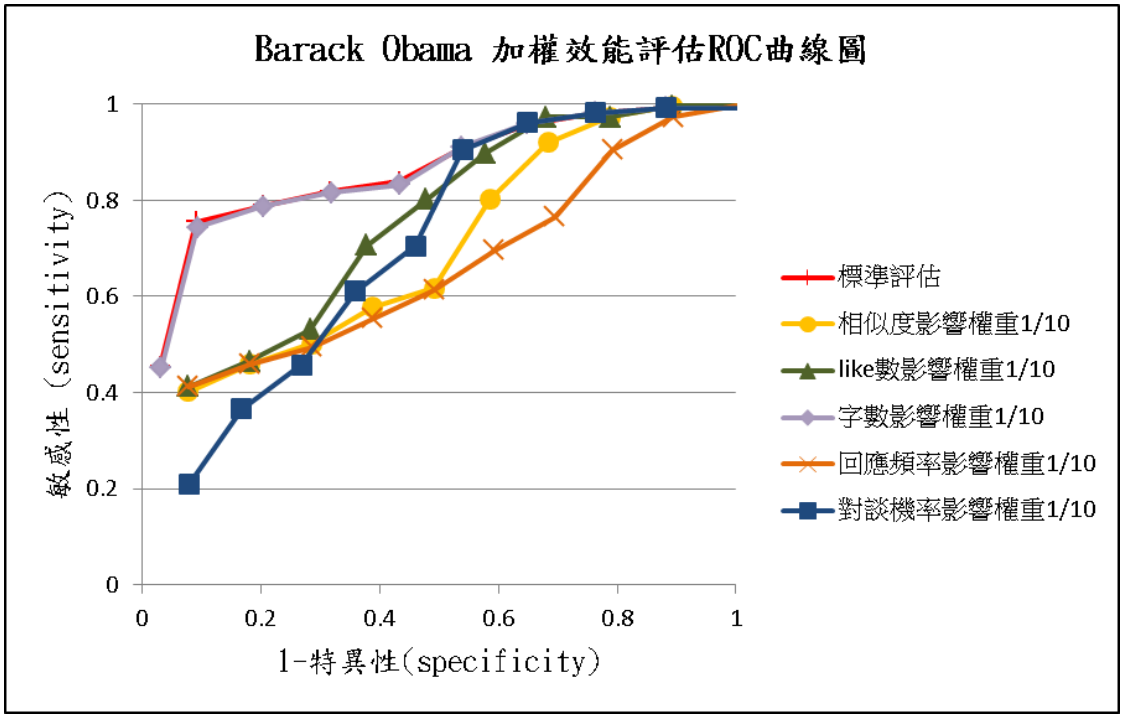

圖 4-3　Barack Obama 粉絲專頁回應加權效能評估 ROC 曲線圖

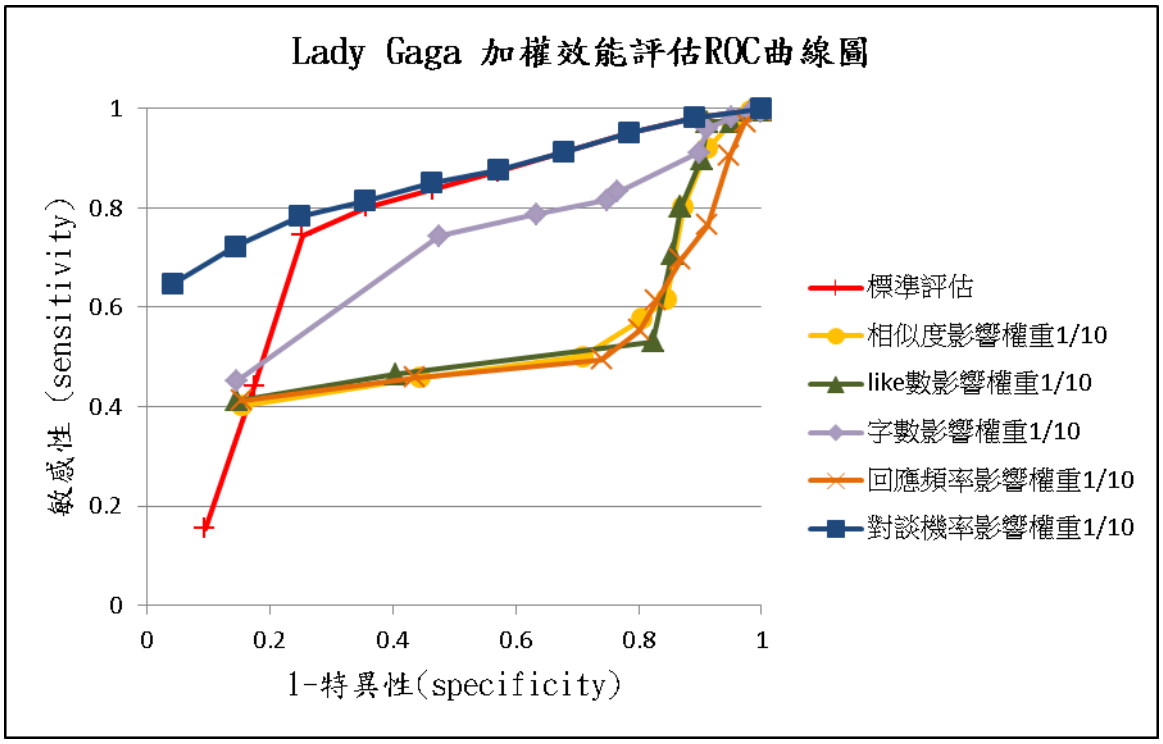

圖 4-4　Lady Gaga 粉絲專頁回應加權效能評估 ROC 曲線圖

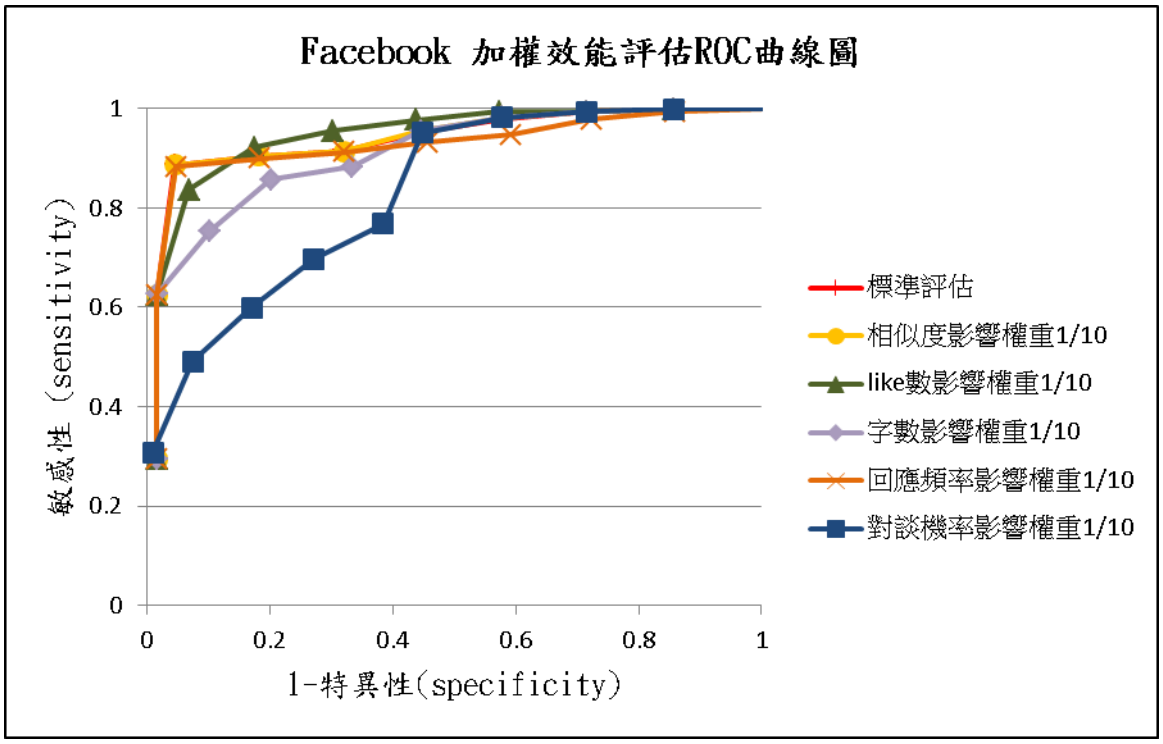

圖 4-5　Facebook 粉絲專頁回應加權效能評估 ROC 曲線圖

從 Obama 的 ROC 曲線圖可以看出，粉絲回應的字數對排名幾乎完全沒有影響，兩條曲線幾乎完全重合，其他屬性對效能都有影響，以敏感性 0.8 的情況來分析，誤判率由大到小依序為回應頻率、文本相似度、對談機率和獲得 like 數，我們也可以說，以本系統對 Obama 的粉絲回應評分，參考屬性重要性的程度以回應頻率最重要，回應字數可以說是毫無參考價值。

ROC 曲線圖中的基準線，是從(0,0)到(1,1)的對角線；基準線上的值就是隨機取樣的機率，在基準線左上的 ROC 曲線，代表效能較隨機(沒有計算模型)時佳；反之，若 ROC 曲線在基準線的右下方，代表這個演算法的效能比隨機還要糟糕。

Lady Gaga 的加權 ROC 曲線圖中，有三條線可算是在基準線的右下方，表示我們用本系統對 Lady Gaga 粉絲回應評分時，不應該忽略文本相似度、獲得 like 數和回應頻率這三個表現的影響，否則結果會比隨機排序還要糟糕。另外，字數多寡對系統效能亦有極大的影響。有趣的是，對談機率的權重降低之後，效能反而變好了，這或許可以解讀為有些粉絲會在專頁上彼此談論和版主發言無關的事。

本系統評分效能最佳的案例，應該是 Facebook 粉絲專頁，在敏感性 0.8 的情況下，只會有 0.04 的誤判率；其中，相似度和回應頻率對排名幾乎完全沒有影響，對談機率則是影響最大的。

表 4-3 是本研究文本集實驗之後得到的相關屬性數據，可以提供此實驗結果更多的佐證和線索。

表 4-3　實驗文本集相關屬性統計表

粉絲專頁名稱	平均相似字數	平均 like 數	平均字數	平均回應次數	回應 1 次以上比例	與人對談比例
Barack Obama	59	0.67	14.17	32.95	67.00%	17.77%
Lady Gaga	18	0.88	5.24	6.86	56.75%	16.53%
Facebook	11	0.42	8.92	3.74	38.62%	18.02%

從表 4-3 可以看出 Obama 粉絲留言的字數和平均相似字數遠高於其他兩個目標，但是從 ROC 曲線圖上可知字數對評估效能完全沒有影響，相似字數卻影響很大，或許我們可以說，留言字數多，相似字數多的機率就變大了，而相似字數對於相關性的判斷幫助很大。就實際觀察文本內容可以發現，可能由於 Obama 的發文多含有專業詞彙，包括經濟用語，法律用語等等，因此粉絲回應較容易使用同樣的詞彙，導致相似度增加，相關度也成正比增加。Obama 粉絲的回應頻率亦遠高於另外兩個目標，粉絲行為較為積極。

Lady Gaga 粉絲的回應行為中，最突出的特徵是獲得的 Like 數，而 like 數確實也是和評估效能非常相關；其他包括回應字數、頻率和相似度也都強烈影響評估效能，綜觀以上數據和觀察原始文本，Lady Gaga 的粉絲發言字數不多，

少數字數較多的回應都是較為相關的內容，發言大部分是針對版主而非其他單一粉絲，產生對談的回應常是與主題無關的內容，例如不論版主發言為何，他們可能在討論串中彼此詢問演唱會或專輯的資訊。

Facebook 的粉絲積極度相對弱得多，其版主發言頻率不高，且發言主題明確，幾乎全為新功能發表或應用說明；粉絲多半針對主題表示看法，與人對談比例雖然高，但平均回應次數低，經常是發表看法或回應一次之後就不再參與。

伍、 結論

本研究希望藉由語意分析、各種 Meta 資料以及文本相似度來協助機器學習對粉絲回應評分，實驗結果確實可能透過這樣的方式獲得正確的篩選。實驗所用的三個目標文本屬性非常不同，一位是政治人物，一位是知名藝人，另一個目標則是一個企業。對於不同屬性的粉絲專頁，我們發現其評分重點不同，對於擁有積極討論粉絲群的專頁，例如政治性的粉絲專頁，只要與人對談且頻率較高，幾乎就可確定是一來一往地針對相關議題討論，相關程度很高；對於偶像崇拜類型的粉絲專頁，無論主題為何，都有簡短的支持口號，相關文本比例較低，機器評分的效能也會較差；至於企業型的粉絲專頁，通常為了宣傳企業作為，粉絲雖然較不積極，但回應的目的性較強，其相關性也較易透過機器評分。

對於 Facebook 以外的各種社群或討論區，也可以利用類似的評分方式，獲得相關性資訊，以解決資訊爆炸時代有效吸取資訊的難題。

參考文獻

1. Cohen, Jacob (1960). "A coefficient of agreement for nominal scales". Educational and Psychological Measurement , Vol. 20, No. 1, pp.37-46.
2. Jaccard, Paul (1901) Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines. Bulletin de la Société Vaudoise des Sciences Naturelles 37, 241-272.
3. Joseph L. Fleiss(1971). "Measuring nominal scale agreement among many raters." Psychological Bulletin, Vol. 76, No. 5, pp. 378–382
4. Kilgarriff and J. Rosenzweig. (2000). English SENSEVAL:Report and Results. In Proceedings of the 2nd International Conference on Language Resourcesand Evaluation, LREC, Athens, Greece.
5. Landis, J.R.; & Koch, G.G. (1977). "The measurement of observer agreement for categorical data". Biometrics Vol. 33, No. 1, pp.159–174.
6. Lesk, Michael. (1986). Automatic Sense Disambiguation: How to tell a pine cone from an ice cream cone. In Proceedings of the 1986 ACM SIGDOC Conference, pp. 24-26, New York.
7. Martin F. Porter(1980) An algorithm for suffix stripping. Program Vol. 14, No. 3, pp.130–137.
8. Purandare and Pedersen (2004) Improving Word Sense Discrimination with Gloss Augmented Feature Vectors. Appears in the Proceedings of the Workshop on LexicalResources for the Web and Word Sense Disambiguation, Puebla Mexico.
9. Roberto Navigli (2009). Word sense disambiguation: A survey. ACM Computing Surveys, 41:10:1–10:69.
10. S. Banerjee, T. Pedersen (2002). An adapted Lesk algorithm for word sense disambiguation using Word-Net.
11. S. Banerjee, T. Pedersen (2003). Extended gloss overlaps as a measure of semantic relatedness.

Online community comments ranking system : Taking Facebook as an example

Yen-Ju Yang[1] Chia-Hung Shen[2]
[1]Department of Information Management, Tatung University yjyang@ttu.edu.tw
[2]Department of Information Management, Tatung University 79912020@ttu.edu.tw

Abstract

With the wide application and diversity access equipment of online community in recent years, people can easily browse and response online information, including photography, text, video and etc.. However, a large part of text response, especially in micro-blogs, without essential meanings except the speaker. Half or more of them are unmeaning words such as greeting, advertising, murmurs, and irrelative arguments. Therefore, people can't catch the main point of respond in short time, and the subject will no longer been concerned and discussed.

For improving this, this study proposes an evaluate system for comments on Facebook-wall posts, and ranking the importance of them. It can help user reading comments high-ranked directly and improve the quality of comments following .

In this study, we classify all comments into 2 sets : (1) relative and (2) irrelative on every stage of : (1) Semantic analysis, (2) Meta-data features compare, and (3) Corpus similarity computing. Base the result, we get TPR(True Positive Rate) = 0.9 for highest as FPR(False Positive Rate) = 0.1.

Keywords: Social network、rank、Lesk Algorithm、Kappa、Facebook.