

國立交通大學

多媒體工程研究所

碩士論文

影評意見探勘及摘要之問答系統

Question Answering - Opinion mining And Auto summarization

for Movie Review

研究生：盧俊錡

指導教授：李嘉晃 教授

中華民國九十八年六月

影評意見探勘及摘要之問答系統

Question Answering - Opinion mining And Auto summarization for
Movie Review

研究生：盧俊錡 Student：Gen-Chi Lu

指導教授：李嘉晃 Advisor：Chia-Hoang Lee

國立交通大學

多媒體工程研究所

碩士論文



A Thesis
Submitted to Institute of Multimedia Engineering
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in
Computer Science
Jun 2009

Hsinchu, Taiwan, Republic of China

中華民國九十七年六月

影評意見探勘及摘要之問答系統

學生：盧俊錡

指導教授：李嘉晃 教授

國立交通大學資訊學院 多媒體工程研究所碩士班



摘要

隨著網際網路的蓬勃發展，部落格與討論版的興起，越來越多人在網路上發表自己對事物的意見以及看法。也因此，當購物者對某件商品下決策時，大部分人們在網路上對該產品的評價往往是很重要的參考依據。例如，一個網路的使用者在選擇要看甚麼電影之前，通常會先瀏覽熱門電影討論版參考看過該電影的使用者的評價來做為決定的因素。但對大部分使用者而言，要消化掉網路上大量對產品的評價資訊可能是相當耗時的。因此，在本篇論文中，我們透過自然語言處理以及資料探勘中的分群技術，來分析影評對該電影的評價是『好看』或『不好看』，並利用自動摘要技巧把影評中『好看』或『不好看』的句子擷取出來回饋給使用者。希望使用者透過我們的介面，可以在比較參考大量的評論資訊時，可以用更簡單，清楚，直覺的比較並做出決定。



Question Answering - Opinion mining And Auto summarization for Movie Review

Student : Gen-Chi LU Advisor : Prof. Chia-Hoang Lee

Institute of Multimedia Engineering

College of Computer Science

National Chiao Tung University



ABSTRACT

With the rapid development of Internet and rise of BLOG and Discussion board , there are more and more people express their views or opinion on things on the internet . Thus , most of people's appraisals on the web are significant information for customer making their decision . For example , people could Decided to go to the movies according to the existing appraisals on the web .But for most of user it is Time consumption to read all reviews on the movie Discussion board .In this Paper ,we apply the Natural Language Processing (NLP) technology and classification technology to classify Text with two polarity : good or bad . Then we combine auto summarization technology to generalized a corresponding appraisal . We hope the user can make decision more rapid through the system which is designed by us.



目錄

中文摘要.....	iv
英文摘要.....	vi
目錄.....	vii
圖目錄.....	viii
表目錄.....	ix
第一章、緒論.....	1
1.1 研究動機.....	1
1.2 研究目的.....	1
1.3 論文架構.....	2
第二章、相關研究.....	3
2.1 情感分類(Sentiment Classification).....	3
2.2 SVM/LIBSVM.....	4
2.3 斷詞與詞性標記.....	4
2.4 自動摘要(Auto Summarization).....	5
2.5 相似詞.....	5
第三章、系統設計.....	8
3.1 概念.....	8
3.2 系統架構.....	8
3.3 影評擷取.....	9
3.4 影評分類階段.....	9
3.4.1 將文章以特徵向量表示.....	10
3.4.2 各種特徵取法.....	10
3.4.3 利用 LIBSVM 訓練出預測模型.....	14
3.5 影評摘要階段.....	14
3.5.1 影評產品特色名詞.....	15
3.5.2 影評正面/負面評論詞.....	16
3.5.3 影評摘要句.....	17
第四章、實驗過程與結果討論.....	18
4.1 交叉測試(Cross Validation).....	18
4.2 實驗結果.....	18
4.3 實驗討論.....	19
第五章、結論與展望.....	20
5.1 研究總結.....	20
5.2 未來研究.....	20
參考文獻.....	21

圖目錄

圖 一 熱門 BBS 站電影討論版	1
圖 二 原始資料分佈	4
圖 三 SVM 找出來的分隔超平面	4
圖 四 Latent Semantic Analysis (LSA) 流程圖。	6
圖 五 字文矩陣	7
圖 六 LSA 降維後的新字文矩陣	7
圖 七 詞的向量	8
圖 八 系統架構	9
圖 九 影評情感分類流程	10
圖 十 移除掉兩個集合交集的詞	11
圖 十一 文章特徵	11
圖 十二 BIGRAM	12
圖 十三 評論摘要流程圖	15
圖 十四 產品特色種子詞擴展	16
圖 十五 負面觀感句子	17

表目錄

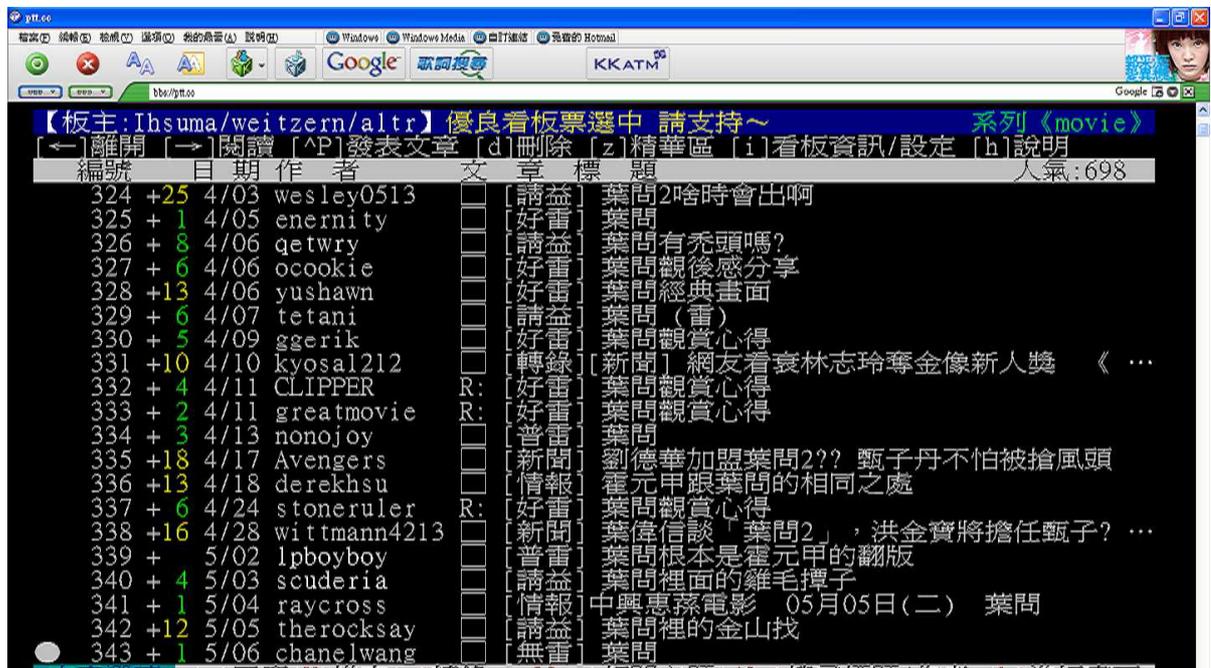
表格 一 產品特色種子詞	16
表格 二 評論詞	17
表格 三 各種特徵判斷情緒的準確度.....	19



第一章、緒論

1.1 研究動機

隨著網際網路的發展帶動部落格以及網路討論版的興起，越來越多人在看完電影以後會去熱門電影討論版或自己的部落格發表對該電影的評價。連帶的，也越來越多人在看電影前會先去熱門電影討論版，或去搜尋部落格，參考這些評價後再決定要看那一部電影。然而網路使用者眾多，同一部電影可能有數十篇甚至上百篇的影評。圖一為國內知名大站 PTT 得電影討論版對電影『葉問』的部分討論文章。



圖一 熱門BBS站電影討論版

在這種資訊量過多的情況下，對使用者而言要去看完每篇影評的評價在去做決定是太過浪費時間的。因此本篇論文希望發展出一套系統，能夠自動的判別出影評的極性是『好看』或『不好看』，能幫助使用者在參考影評時節省時間。

1.2 研究目的

近幾年關於意見探勘(OPINION MINING)的研究討論越來越多，本篇論文將應用意見探勘的技巧，把電影影評分為兩類：『好看』以及『不好看』。而本論文系統可以將使用者想要參考的相關電影影評集分成兩類，找出大部分人對該電影的

評價看法；再利用摘要的技術分別對兩類影評做摘要。不僅讓使用者透過本系統知道大部分人對該電影的評價，並且將擷取出的評價摘要回饋給使用者，以下兩句話 S1 和 S2 為例：

S1:真心推薦這一部讓你很感動很感動台灣人的電影！

S2: 不必期待有什麼賺人熱淚~高潮迭起的畫面出現。

藉此讓使用者能迅速的知道為什麼評價為『好看』的原因為何，評價為『不好看』的原因為何，以利於使用者基於自己的喜好做判斷與決定。

1.3 論文架構

第一章：前言，描述本文的研究動機、目的，將本文系統的初衷和基本概念做一個介紹。

第二章：相關研究，說明意見探勘(OPINION MINING)中關於情感分類(Sentiment Classification)的研究以及摘要和其他相關技術(斷詞、SVM、同義詞等)的研究。

第三章：系統設計，將前置作業和系統的整體架構做一個完整介紹。

第四章：實驗過程與結果討論，將實作出來的系統做一些比較與討論。

第五章：系統的結論與展望，將系統做總結和探討系統未來方向。

第二章、相關研究

2.1 情感分類(Sentiment Classification)

在 [1] 論文中用了一種非監督式的方法去計算文件每個字和『excellent』或『poor』的相互關係，即 PMI，來計算一個字的極性分數，而利用這些字的極性分數可以判別出該文件的極性。

[2] 論文則介紹一以上篇論文為基底的評論探勘系統『OPINE』。[3] 則以上述技術延伸發展網路意見特徵評價應用在旅館評論比較上。

前述三篇研究基本流程都是先找出評論產品特色(product feature)：例如一篇討論數位相機的評論中，常討論的數位相機產品的特色有：畫素、光圈、電池續航力等，而這些產品特色通常是評論作者決定該產品好或不好的重要依據。而評論詞(opinion word)通會出現在這些產品特色詞附近，用來評論這些產品特色的好或壞：例如畫素附近可能出現清新，模糊等評論詞。藉由產品特色和評論詞這樣的 pair(ex. 畫素，清新)來找出評論作者對產品的感想的評論句。藉由算出評論詞的極性來判斷影評的屬於正面或負面。計算極性的方法如下：

- Semantic Orientation(SO):

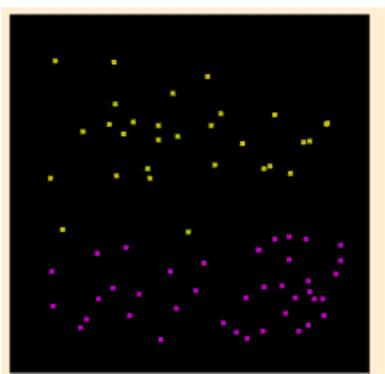
$$PMI(word_1, word_2) = \log_2 \left[\frac{p(word_1 \& word_2)}{p(word_1) p(word_2)} \right]$$

$$SO(phrase) = PMI(phrase, "excellent") - PMI(phrase, "poor")$$

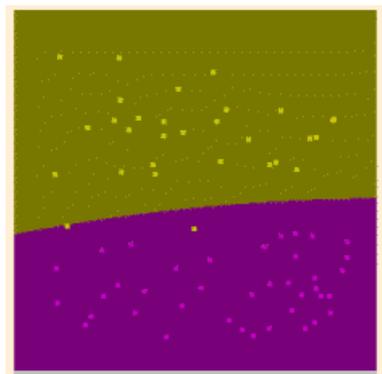
在 [4] 中則介紹了應用各種機器學習(machine learning)的方式訓練資料，諸如『Naïve Bayes』、『Maximum Entropy』、以及『Support Vector Machines(SVM)』將情感分類。

其中 SVM 被廣泛應用在文件分類，效果也普遍不錯。[5] 和 [6] 分別也應用 SVM 對文件做分類。本論文將著重在應用 SVM 來做情感分類。

2.2 SVM/LIBSVM



圖二 原始資料分佈



圖三 SVM 找出來的分隔超平面

SVM 是現今被廣為應用在分類(Classification)的技術。如果我們有一組已經訓練好的資料， $\{X_1 \dots X_n\}$ ， $X_i \in \mathbb{R}^d$ ， $1 \leq i \leq n$ 。同時我們把這組資料裡的每筆資料標上標記 $\{Y_1 \dots Y_n\}$ ， $Y_i \in \{-1, 1\}$ 。則根據這組標記好的資料，SVM 會畫出超平面(hyperplane)將這些散佈在 d 為空間的點分成兩組，如圖二和圖三。因此當有一筆新的未分類資料 X_k 進來時，只要根據該資料在 d 維空間中的位置座落在超平面的那一邊而預測他的標記 Y_k 值。

本文中 SVM 分類的實做使用台大林智仁老師所開發的 LIBSVM[7]，一套 SVM 工具，透過 LIBSVM 所訓練的分類數學模型整合入本系統中，借此判斷分類文章的情感。

2.3 斷詞與詞性標記

斷詞與詞性標記是自然語言處理中基礎且重要的一部份，機器翻譯、資訊擷取、摘要製作及自動作文評分系統等研究都需利用斷詞及詞性標記處理後的結果來進行下一步動作，故斷詞的結果的正確率對研究成果有直接影響。

在中文的句子中，通常不存在有空白這個單元，所以不像英文的句子可以分得很清楚哪邊為一個字，哪邊為一個詞，所以我們藉助中央研究院的詞庫小組中文斷詞系統[8]來做斷詞與詞性標記的工作，其正確率可達到 95~96%之間。

2.4 自動摘要(Auto Summarization)

自動摘要的作法，大抵可分為『摘錄』(extraction)與『摘要』(abstraction)兩種。『摘錄』的結果為文件中重要文句的重組，其作法比較不依賴額外的知識或資源，主要是根據使用者的需求，從文件本身或其他相關的文件中選取重要文句，編輯組合成使用者預期的長度即可。 [9]運用文法鍊結(Lexical Chains)將文章中的句子分群，再從每個重點群中各挑出一句代表的句子做為摘錄結果。相對摘錄，『摘要』的結果則不限於文件中的文句，其作法需要較多人工準備的資源，如辭典、同義詞庫、詞性標記、語法樹等，經自然語言處理後，自動生成涵蓋原文重點的簡潔文句。 [10]中即是以摘錄的方法，使用大量的人工摘要做為資料庫、文法字典等，對文章原文句子做刪減產生摘要。如下：

原文：

When it arrives sometime next year in new TV sets, the V-chip will give parents a new and potentially revolutionary device to block out programs they don't want their children to see.

摘要：

The V-chip will give parents a device to block out programs they don't want their children to see.

由於「摘要」所需資源較多，目前以「摘錄」為主要的研究佔較多數。

另外在[11]中提出了能根據使用者興趣和需求而做出相對應的偏重摘要研究。比起一般摘要要是擷取出文章作者主觀的重點句，偏重摘要能摘出更符合使用者想知道的資訊的摘要。

2.5 相似詞

[12]中應用了潛藏語意分析(Latent Semantic Analysis)的矩陣模型來計算文件資料庫中詞語詞之間的相似程度。流程如圖四。

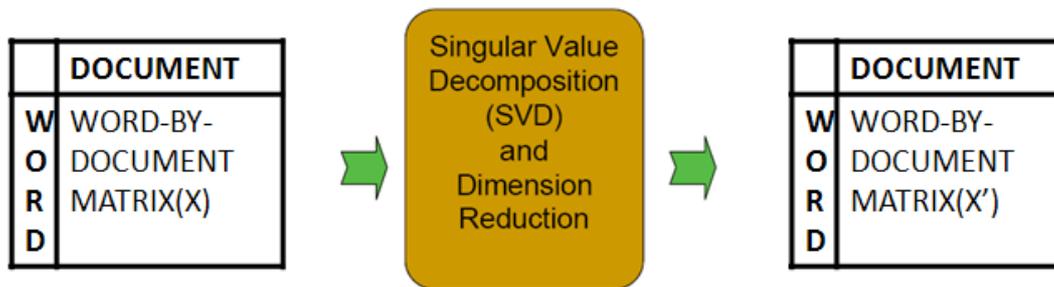


圖 四 Latent Semantic Analysis (LSA)流程圖。

例如一個文件資料庫共有9篇文章如下(紅體字為關鍵字)

- d1: Human machine interface for ABC computer application
- d2: A survey of user opinion of computer system reponse time
- d3: The EPS user interface management system
- d4: System and human system engineering testing of EPS
- d5: Relation of user perceived reponse time to error measurement
- d6: The generation of random, binary, ordered trees
- d7: The intersection graph of paths in trees
- d8: Graph minors IV: Widths of trees and well-quasi-ordering
- d9: Graph minors: A survey

我們可以建立一個字文索引矩陣如圖五。經過LSA降為計算後我們可以得到新的字文索引圖六。透過LSA，我們可以找出詞和文章間更深層的關係。像『tree』這個詞沒有出現在文件9裡面，原本的字文矩陣我們找不到tree和文件9的關係。但『tree』和『minor』同時出現在文件8，而『minor』出現在文件9，經過LSA降維運算以後我們找到了『tree』和文件9的間接關係。從新的字文矩陣中，我們可以把詞和每篇文章的關係以向量表示，如圖七。如此每個詞的相似程度即為兩個向量的內積。如下式：

$$\text{SIM}_{w_i, w_j} = \frac{w_i \cdot w_j}{|w_i| |w_j|}$$

	D1	D2	D3	D4	D5	D6	D7	D8	D9
Human	1	0	0	1	0	0	0	0	0
Interface	1	0	1	0	0	0	0	0	0
Computer	1	1	0	0	0	0	0	0	0
User	0	1	1	0	1	0	0	0	0
System	0	1	1	2	0	0	0	0	0
Response	0	1	0	0	1	0	0	0	0
Time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
Survey	0	1	0	0	0	0	0	0	1
Trees	0	0	0	0	0	1	1	1	0
Graph	0	0	0	0	0	0	1	1	1
Minors	0	0	0	0	0	0	0	1	1

圖 五 字文矩陣

	D1	D2	D3	D4	D5	D6	D7	D8	D9
Human	0.16	0.40	0.38	0.47	0.18	-0.05	-0.12	-0.16	-0.09
Interface	0.14	0.37	0.33	0.40	0.16	-0.03	-0.07	-0.10	-0.04
Computer	0.15	0.51	0.36	0.41	0.24	0.02	0.06	0.09	0.12
User	0.26	0.84	0.61	0.7	0.39	0.03	0.08	0.12	0.19
System	0.45	1.23	1.05	1.27	0.56	-0.07	-0.15	-0.21	-0.05
Response	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
Time	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
EPS	0.22	0.55	0.51	0.63	0.24	-0.07	-0.14	-0.20	-0.11
Survey	0.10	0.53	0.23	0.21	0.27	0.14	0.31	0.44	0.42
Trees	-0.06	0.23	-0.14	-0.27	0.14	0.24	0.55	0.77	0.66
Graph	-0.06	0.34	-0.15	-0.30	0.20	0.31	0.69	0.98	0.85
Minors	-0.04	0.25	-0.10	-0.21	0.15	0.22	0.50	0.71	0.62

圖 六 LSA降維後的新字文矩陣

	D1	D2	D3	D4	D5	D6	D7	D8	D9
Human	0.16	0.40	0.38	0.47	0.18	-0.05	-0.12	-0.16	-0.09
Interface	0.14	0.37	0.33	0.40	0.16	-0.03	-0.07	-0.10	-0.04
Computer	0.15	0.51	0.36	0.41	0.24	0.02	0.06	0.09	0.12
User	0.26	0.84	0.61	0.7	0.39	0.03	0.08	0.12	0.19
System	0.45	1.23	1.05	1.27	0.56	-0.07	-0.15	-0.21	-0.05
Response	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
Time	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
EPS	0.22	0.55	0.51	0.63	0.24	-0.07	-0.14	-0.20	-0.11
Survey	0.10	0.53	0.23	0.21	0.27	0.14	0.31	0.44	0.42
Trees	-0.06	0.23	-0.14	-0.27	0.14	0.24	0.55	0.77	0.66
Graph	-0.06	0.34	-0.15	-0.30	0.20	0.31	0.69	0.98	0.85
Minors	-0.04	0.25	-0.10	-0.21	0.15	0.22	0.50	0.71	0.62

圖 七 詞的向量

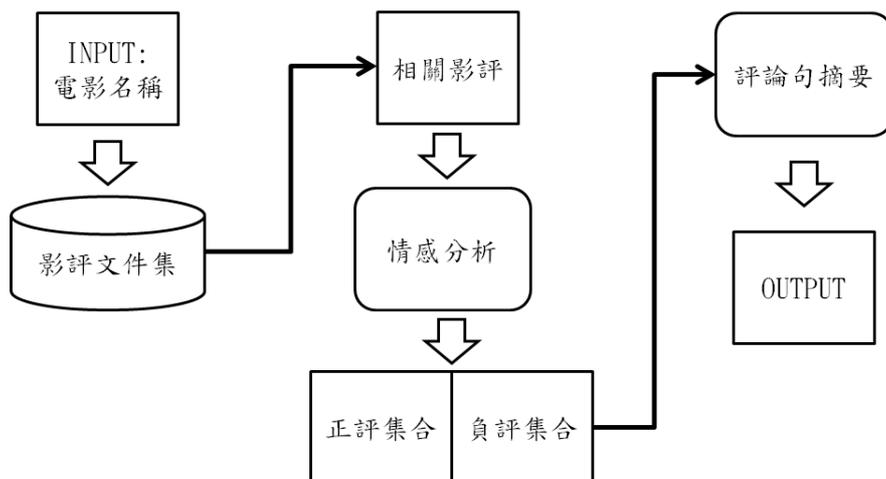
第三章、系統設計

3.1 概念

本篇論文中，我們將試著利用各種文章的特徵(feature)將文章的情感分類成『好評』或『負評』，再根據分類完的結果，分別對『好評』文章集合與『負評』文章集合摘出重要句子，試著找出原文作者覺得電影好看或不好看的原因，並將此回饋給使用者。

3.2 系統架構

本系統共分為三個部分。第一部分影評擷取中根據使用者輸入的電影名稱到文件資料庫中收集相關的電影影評集合。在第二部分影評分類中，我們將從第一部分找到的所有相關文章分成兩類，『好評』和『負評』。在第三部分影評摘要中，我們將各自對好評文章集合和負評文章集合中擷取中重要的摘要句回饋給使用者，讓使用者可以根據這些擷取出來的資訊迅速的做出判別和決定。圖八為整體系統架構。



圖八 系統架構

3.3 影評擷取

本系統事先收集數百篇電影影評，經由 CKIP 斷詞完以後，存在系統資料庫中。每篇影評檔名皆以電影名稱命名儲存。根據使用者輸入的電影名稱，系統會到資料庫中找檔名和輸入名稱相符的影評，並將之收集起來，在下一步『影評分類階段』做進一步的處理。

3.4 影評分類階段

以下介紹影評情感分類的部分。圖九為影評情感分類的流程圖。其中訓練資料為國內熱門 BBS 站 PTT(telnet://ptt.cc)電影討論版所收集的好評和負評文章各 250 篇。接下來我們會介紹如何將文章以特徵向量表示，以及各種特徵的取得。

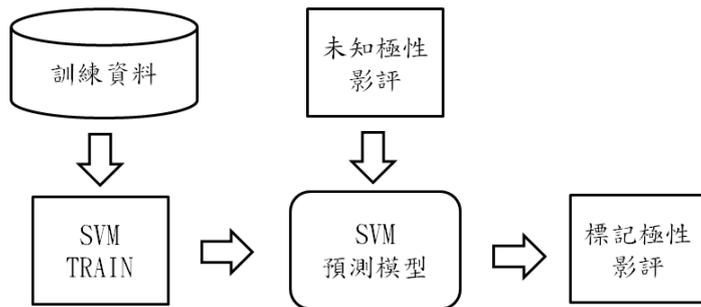


圖 九 影評情感分類流程

3.4.1 將文章以特徵向量表示

假定我們有一組 feature $F1$ {『好看』,『沉悶』,『精彩』,『冗長』}, 我們可以出現該 feature 與否, 表示一篇文章以向量表示。如一篇文章 T1 如下:

『這一部電影相當**好看**, 景觀寬闊, 分鏡優美, 打鬥畫面也相當**精采**, 不愧是大卡司的巨作!**好看!**』

T1 以 $F1$ 這組特徵表示為 $\langle 1_{好看}, 0_{沉悶}, 1_{精彩}, 0_{冗長} \rangle$ 。

依此規則, 我們將以分類訓練好的影評資料正評/負評各兩百五十篇, 根據特徵以向量表示, 以利於之後 LIBSVM 工具訓練預測模型(predict model)。

3.4.2 各種特徵取法

本小節中, 我們將介紹各種文章特徵, 而針對這些特徵所做出來的預測模型的準確率, 將在第四章的實驗結果討論。我們先將訓練資料中所有形容詞、動詞、感嘆詞、副詞等比較能代表人類情感的詞曲出來, 並定義兩種集合如下:

$$pword = \{word\ w_1 | w_1 \subseteq pos\ text\}$$

$$nword = \{word\ w_1 | w_1 \subseteq neg\ text\}$$

第一組實驗文章特徵:

1. 先將兩個集合交集的字移除掉, 如圖十所示。

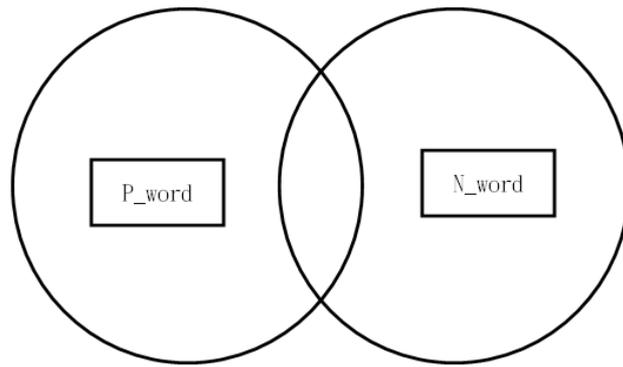


圖 十 移除掉兩個集合交集的詞

2. 將出現頻率過少的詞移除掉，僅保留高頻率詞。高頻率詞定義如下：

$f(w)$: 詞 w 出現次數

$$\text{sum_f} = \sum f(w)^2$$

$$\text{avg} = \text{sum}/n$$

詞 w 為高頻率詞, if $f(w)^2 > \text{avg}$



用此方法，我們在 250 篇正評中找到 1127 個文章特徵，在 250 篇負評中找到 222 個文章特徵。圖十一和圖十二為找到的文章特徵。

聯手(VH)	大失所望(VH)
諒解(VK)	武俠(A)
明快(VH)	無雙(VH)
每每(D)	記載(VC)
雙面(A)	萬惡(A)
不休(VH)	圍攻(VC)
堆砌(VC)	愚痴(VH)
揭露(VC)	亡國(VH)
秉持(VJ)	不妙(VH)
繽紛(VH)	暴力(VH)
猜疑(VE)	為什嘛(VH)
靜靜(VH)	擋下(VC)
悲痛(VH)	端出(VC)
契合(VH)	詭辯(VA)
平常(VH)	辯論(VE)
微不足道(VH)	不倫不類(VH)
奠定(VC)	轉變(VA)
天真爛漫(VH)	調情(VB)
有理(VH)	搞出(VC)
中規中矩(VH)	草草(D)

圖 十一 文章特徵

我們接著再藉由這些找出來的詞，到訓練資料查看這些詞前後常伴隨著那些字出現，即對每個詞找出他們常出現的 BIGRAM。例如在我們訓練資料中『精彩』這個字前後可能常常出現『非常精彩』或『精彩萬分』等 BIGRAM。常出現的 BIGRAM，也就是高頻率的 BIGRAM，定義如上述。

我們在 250 篇正評中找到 316 個 BIGRAM，在 250 負評文章中找到 237 個 BIGRAM

。圖十二為找到的 BIGRAM。

充滿	歡笑	9	白白	浪費	16
置身	其中	16	白白	逃掉	4
悄悄	離開	4	前途	無量	4
慶幸	自己	9	功德	無量	4
孤寂	執迷	4	活捉	作為	4
扣除	頂上	4	活捉	知情	4
後設	電影	4	殲滅	所有人	4
慘澹	經營	4	因為	打擊	4
關上	奮戰	4	臆測	浮現	4
療傷	治癒	4	燕飛兒	翻白眼	4
放聲	大哭	36	翻白眼	落馬	16
輸給	無間道	4	醜化	陳慧琳	4
無間道	系列	4	醜化	亞洲人	4
畢生	收藏	4	呻吟	連連	16
難得	正經	4	呻吟	一下	4
親生	女兒	64	燕飛兒	呻吟	4
仇人	親生	4	呻吟	難耐	4
不會	輸給	9	豬頭	寫出來	4
填補	自己	4	預先	排練	4
已經	逝去	4	主角	預先	4

圖 十二 BIGRAM



第二組實驗文章特徵：

在前敘的方法中，我們移除掉這些同時出現在這評和負評的字詞。在此組實驗特徵中，我們不移除這些字，即是對五百篇影評資料(正評/負評文章各 250 篇)取出高頻率的字詞和字詞 BIGRAM。

第三組實驗文章特徵：

[5]中對於文章特徵的取法是在訓練資料中出現頻率大於 3 次且不為常出現用詞(ex. 似乎，但是，因為…等詞)的詞即為文章特徵。為了比較高頻率的詞和 BIGRAM 以及出現超過三次以上的詞和 BIGRAM 兩種文章特徵的好壞，這組實驗文章特徵中，只要在訓練資料中出現頻率大於 3 的詞和詞前後文皆列為文章特徵。

其他特徵一：

一個詞前面是否出現否定詞會影響該詞的極性，如『好看』和『不好看』。我們假設一個詞前面是否有出現否定詞會影響該詞的極性，並同實影響對文章情感的判斷。所以我們把一個字詞或字詞前後文前面是否有出現否定詞也獨立成一個特徵。例如現在我們有一組特徵 $F1$ {『好看』,『沉悶』,『精彩』,『冗長』}，加入否定詞以後該組特徵會變成 $F1_{\text{否定}}$ {『好看』,『否定好看』,『沉悶』,『否定沉悶』,『精彩』,『否定精彩』,『冗長』,『否定冗長』}。在這種情況下，如下列兩篇文章：

T1:『這一部電影相當**好看**，景觀寬闊，分鏡優美，打鬥畫面也相當**精采**，不愧是大卡司的巨作！**好看**！』

T2:『這部電影一點都**不好看**，**沒有**那麼**精彩**，去電影院看真是浪費錢！』

在 $F1$ 特徵中，兩篇文章的特徵表示向量均為 $\langle 1_{\text{好看}}, 0_{\text{沉悶}}, 1_{\text{精彩}}, 0_{\text{冗長}} \rangle$ 。但加入否定詞後用

$F1_{\text{否定}}$ 這組特徵，兩篇文章的特徵表示向量為

$\langle 1_{\text{好看}}, 0_{\text{否定好看}}, 0_{\text{沉悶}}, 0_{\text{否定沉悶}}, 1_{\text{精彩}}, 0_{\text{好看}}, 0_{\text{冗長}}, 0_{\text{否定冗長}} \rangle$ 和

$\langle 0_{\text{好看}}, 1_{\text{否定好看}}, 0_{\text{沉悶}}, 0_{\text{否定沉悶}}, 0_{\text{精彩}}, 1_{\text{否定精彩}}, 0_{\text{冗長}}, 0_{\text{否定冗長}} \rangle$ 。

其他特徵二：加入字數

在觀察中我們察覺到，好評通常文章字數比較多，這可能可以歸咎在人在寫自己喜歡的事物的時候會比較願意花時間在撰寫和描述這事物，而對討厭的事物則否。不管實際上原因是否如此，但根據我們對五百篇影評做的統計，好評長度平均為一千個字，負評長度平均六百個字。因此我們假設文章的字數可能也是會影響文章情感判斷的一個特徵。在此，我們將文章字數分為四個特徵：

{0~500, 500~1000, 1000~1500, 1500~2000, 2000 以上}。

其他特徵三：加入位置

我們假設一個詞出現在文章的位置也可能會是影響這篇文章情感的重要特徵。例如，通常放在文章最後一句的字詞同時代表了這篇文章的結論，而放在文章中段的字詞可能只是討論劇情。因此我們試著把字詞的位置也加入特徵。

例如一組特徵 $F1$ {『好看』, 『沉悶』, 『精彩』, 『冗長』}，我們把字詞出現位置定義為出在文章前半段和文章後半段，則此組特徵會變成

$F1_{位置}$ {『好看(前)』, 『好看(後)』, 『沉悶(前)』, 『沉悶(後)』, 『精彩(前)』, 『精彩(後)』, 『冗長(前)』, 『冗長(後)』}。一篇文章如下：

T1: 『這一部電影相當**好看**，景觀寬闊，分鏡優美，打鬥畫面也相當**精采**，不愧是大卡司的巨作！**好看**！』

用 $F1$ 特徵表示向量為 $\langle 1_{好看}, 0_{沉悶}, 1_{精彩}, 0_{冗長} \rangle$ ，用 $F1_{位置}$ 則為

$\langle 1_{好看前}, 1_{好看後}, 0_{沉悶前}, 0_{沉悶後}, 0_{精彩前}, 1_{精彩後}, 0_{冗長前}, 0_{冗長後} \rangle$ 。

在本論文中，我們將字詞出現的位置定義成三個，出現在前段(文章前 1/3)，中段(1/3 到 2/3)，出現在後段(文章後 1/3)。

3.4.3 利用 LIBSVM 訓練出預測模型

我們根據以上特徵將已分類好訓練資料用 SVM 工具，LIBSVM 訓練出一個預測模型，之後任何一篇未分類的文章，我們用同一組特徵值將文章表示成向量以後，便可用這預測模型幫未知的文章分類。

3.5 影評摘要階段

和一般摘要不同的是，我們並不是要挑選出最重要的句子，而是要找出影評作者覺得電影好看或不好看的原因。例如：

S1：讓整部鳳凰會的密令從頭到尾節奏明快、敘事完整，

S2：這片可以說真的相當無趣，精神不好的人大概會看到睡著，

如此使用者可以知道影評作者覺得好看或不好看的原因，並藉此迅速的做出判斷決策。圖十三為評論摘要流程圖。

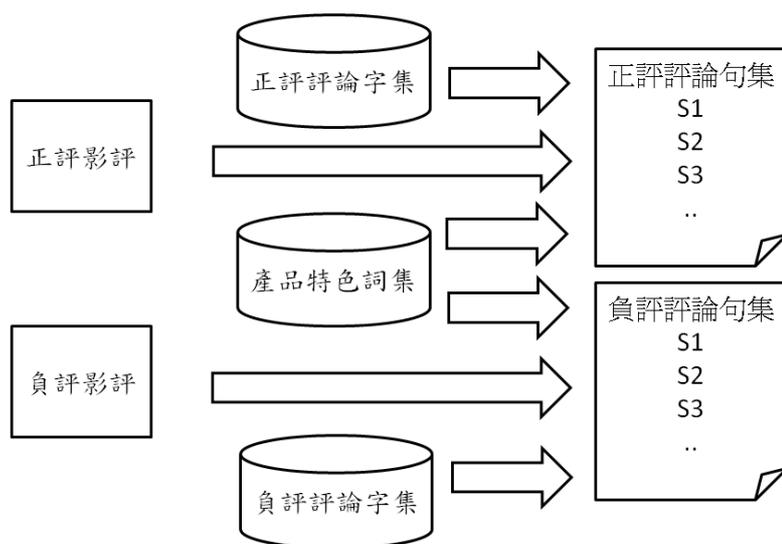


圖 十三 評論摘要流程圖

3.5.1 影評產品特色名詞

首先，我們必須先找出一般影評文章主要在討論電影好不好看的產品特色名詞有那些，例如導演、演員、卡司、特效等。而影評中出現這些產品特色名詞的句子有很大的可能是該影評評論一部電影好看或不好看的重點。例如產品特色名詞『特效』可能出現在『**變形金剛電影特效相當精彩，讓人看得直呼過癮。**』這種強烈表達影評作者為什麼覺得電影好看的重點句。

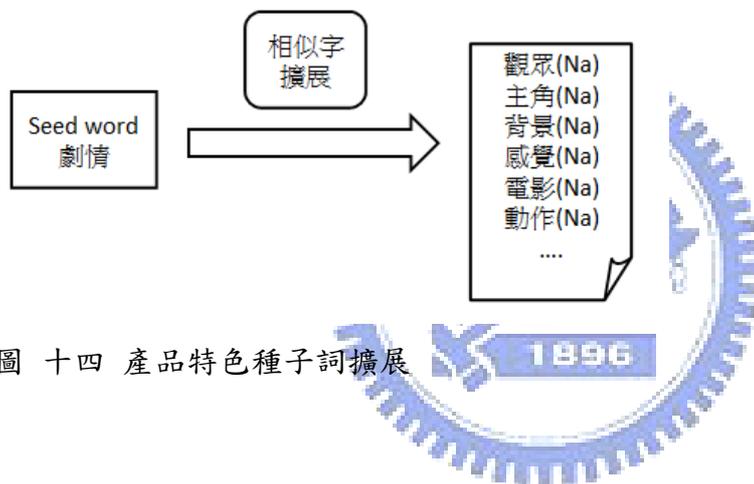
[1][2][3]中對產品名詞的取法並沒有清楚詳敘，但大抵上是取訓練資料中常出現的名詞及名詞片語為產品特色名詞。在我們的系統中，我們先用人工定義 10 個人們在寫影評時常會用來評斷一部電影好看或不好看的重點，當做電影影

評的產品特色名詞的種子。這些詞如表格一。

表格一 產品特色種子詞

場景 劇情 特效 演技 動作 動畫 音樂 導演 結局 運鏡

藉由這些種子詞，我們運用 2.5 介紹的方法：使用 LSA 來計算兩個詞間的相似度，分別去計算訓練資料中每個名詞對這些種子詞的相似度分數。每個種子詞在取前 1%和它最相似的名詞做為擴展。圖十四為種子詞『劇情』擴展的流程以及結果。



圖十四 產品特色種子詞擴展

這些擴展出來的名詞集合以及原本的產品特色種子，即為本系統對於電影影評的產品特色名詞。藉由找出影評含有這些詞的句子，我們可以找出影評作者討論一部電影好看或不好看的重點句子。

3.5.2 影評正面/負面評論詞

根據前小節，我們得到產品特色名詞，可以找到影評作者在討論一部電影好看或不好看的重點句；接下來我們要用評論詞(OPINION WORD)來替這些句子標記上極性為好看或不好看。

從五百篇訓練資料中，我們用人工的方法挑選出 497 句帶有強烈正面觀感的句子以及 361 句帶有強烈負面觀感的句子。圖十五為我們挑選出來的負面觀感句

子。

Goldenberg會莫名其妙地插入一些不僅與電影其他部份相當不搭調，本身素質也不甚突出的對白。

這些對白本身不真實且有些可笑不說，也拖慢了電影的節奏，讓故事出現了一些斷層。

Goldenberg仍沒有讓電影有太多深度：

但整體故事仍有種些微的流水帳感，

一開始看的時候，感覺起來就很不順暢。

照著某某角色的臉的特寫，可是感覺就還不到兩秒，

畫面就切換到別的地方去；感覺就很突兀，

在劇情交待上，也會有那種不到兩秒感覺很不順暢，很不夠味。

前半段的配樂真的有夠爛的。感覺就像是打電動會聽到的那種音樂 ...。

浪費了甄子丹的一身功夫。

《畫皮》的武打是另一項我很想鞭的東西。

圖 十五 負面觀感句子

從這些句子中，我們分別從正面觀感的句子和負面觀感的句子中挑選出高頻率的形容詞、動詞、感嘆詞以及 BIGRAM 當作描述影評產品特色的評論詞，高頻率的定義如同 3.4.2 小結所敘。找到的字詞以集字詞前後文如表格二

表格 二 評論詞

	正面	負面
評論詞	清晰，呈現，展現出，華麗 ...	莫名，突兀，草率，鬆散 ...
BIGRAM	清新 明快，華麗 多變 ...	劇情 空洞，不合 邏輯 ...

3.5.3 影評摘要句

根據 3.4 小節，我們可以將一篇影評分好類『好評』或『負評』。我們先取出影評所有包含有產品特色詞的句子，依據該篇影評分類的結果，我們取相對應

極性的評論詞做摘要。換句話說，如果一篇影評被歸為好評，影評中出現特色名詞的句子中，挑出同時也出現正面觀感情緒詞的句子為摘要句；而反之亦然。在摘要過程中，一個評論詞的極性(正面或負面)會隨著該詞前面是否出現否定詞而轉換，例如：『好看』本來屬於正面觀感評論詞，而『不』『好看』則為負面觀感評論詞。

第四章、實驗過程與結果討論

4.1 交叉測試(Cross Validation)

在本節中我們將對 3.4.2 小節中所提出的各種特徵做情感分類的準確度比較。計算準確度的方法我們採用交叉測試，步驟如下：

1. 先有已分好類的一堆資料
2. 隨機分成 n 組
3. 挑其中一組為訓練資料其它組為測試資料

並計算其準確率

4. 輪流對每一組資料重複 3 的動作
5. 計算平均準確率

4.2 實驗結果

實驗資料：國內知名 BBS 站 PTT(telnet://ptt.cc)電影討論版 470 篇影評(正評 250 篇/負評 220 篇)。表格三為各組交叉測試實驗結果。



表格 三 各種特徵判斷情緒的準確度

	Test
第一組特徵	71.0021
第一組特徵+否定	70.7889
第一組特徵+位置	69.5096
第一組特徵+長度	70.1493
第二組特徵	78.4648
第二組特徵(否定)	79.3177
第二組特徵(位置)	71.6418
第二組特徵(長度)	79.1045
第三組特徵	76.5458
第三組特徵(否定)	75.4797
第三組特徵(位置)	70.1493
第三組特徵(長度)	78.2516

4.3 實驗討論

參照各組加入否定、位置、長度等特徵以後，其準確度效果均無明顯提升或下降，表示這些特徵對文章分類並沒有太大的影響力，這實驗結果和[1]的實驗結論雷同。而和[2]一樣的方法取文章特徵(第三組特徵)相比較，第二組實驗效

果有較佳的結果。根據實驗結果，本系統取第二組特徵加上否定詞做訓練預測模型的文章特徵。

第五章、結論與展望

5.1 研究總結

透過上述研究，我們實做了一套可以判別評論意見的系統，透過這系統我們可以更迅速的分析網路上大量的評論並得到精簡的意見摘要。從這些摘要我們可以花費較少的時間去得知大部分人的意見並從中判斷。從實驗結果來看，這方法是可行的，並且可以應用在更多地方；例如企業新產品的使用心得分析等。

5.2 未來研究

情感分類部分，對文章特徵，或許可以找到更好的特徵或特徵表示法使 SVM 能訓練出準確率較高的預測模型。或者也可以結合 SVM 和其他情感探勘的技巧加強情感的分析。

摘要方面，如何找出更精確的特色名詞以及情緒詞也可以達到更棒的意見摘要。

參考文獻

- [1] Peter D. Turney. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews . Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, July 2002, pp. 417-424.
- [2] **Ana-Maria Popescu and Oren Etzioni. Extracting Product Features and Opinions from Reviews.** In proceedings of HLT-EMNLP'2005.
- [3] 陳正揚. Feature Appraisal for Hotel Comparison. 2006.
- [4] Bo Pang and Lillian Lee , Shivakumar Vaithyanathan . Thumbs up? Sentiment Classification using Machine Learning Techniques. Proceedings of EMNLP 2002, pp. 79–86.
- [5] Thorsten Joachims . Text Categorization with Support Vector Machines : Learning with Many Relevant Features. 1998.
- [6] 楊昌樺, 高虹安, 陳信希. 以部落格語料進行情緒趨勢分析. 2007.
- [7] 林智仁. <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>
- [8] <http://ckipsvr.iis.sinica.edu.tw/>
- [9] Regina Barzilay , Michael Elhadad. Using Lexical Chains for Text Summarization. In: Mani, Inderjeet and Maybury, Mark T. *Advances in automatic text summarization*, Cambridge, MA: The MIT Press, pages 111–121. 1999.
- [10] Hongyan Jing. Sentence Reduction for Automatic Text Summarization. 2000.
- [11] 關英杰, 林鴻飛, 楊志豪, 趙晶. 關鍵詞密度分佈法在偏眾摘要中的應用研究. 中國期刊第33卷16期, 人工智能及識別技術. 2007.
- [12] Yi-Fen Chen, Automatic Concept-Based Answer-Finding System . 2003 .