基於查詢結果與特徵推導之廠商及產品關聯推斷技術

Business & Product Relation Inference based on Query Result and Feature Derivation

許國信
國立中央大學資工系
david35004@gmail.com

莊秀敏
國立中央大學資工系
showmin1205@gmail.com

張嘉惠
國立中央大學資工系
chia@csie.ncu.edu.tw

摘要

隨著產業發展更替和生產技術的提升，公司與其製造生產的產品資訊經常在變動，且產品(材料亦為一種產品)間也有其上下游或平行對應關係，而這些資訊大多倚靠專家整合，不只耗時也耗力。如何透過 Web 上大量的資源或產業新聞，經由機器學習的方法建立分類模組，進而判斷關聯為一大挑戰。

我們利用 Web 蒐集到的資訊和產業新聞，擷取十五個特徵來建立分類模組，藉此判斷該公司是否有生產該產品，得到準確率(accuracy)為 0.844；產品與產品間則透過十六個特徵去判斷是上下游或平行關係，準確率為 0.742。

關鍵詞：分類模組、機器學習、特徵擷取

1. 緒論

面臨全球景氣變化挑戰，產業鏈隨時都在更替，掌握供應鏈的關係能迅速聯繫替代公司或相關公司，減少因變動產生的衝擊，而早一步做出決策，便能取得優先權，進而提高獲利。

為了強化國內產業上中下游之結構，建立自主供應體系，需要進行產業分析。過去產業分析需要閱讀大量產業新聞，不斷累積經驗才能判定公司與產品或產品間的關聯，因此需要培養相關專業人士。

本論文主要在解決以下兩種關聯問題(1)公司與產品是否有關聯，即該公司是否有生產該產品。(2)產品一與產品二為上下游關係或平行關係。

而過去倚靠專業人工去處理資料，無論是手稿或是運用文書軟體記載都過於耗時。因為沒有固定方式儲存，容易造成遺失，整理上亦不方便。而工商記錄登記資訊並不完善，更新速度也較慢，因此本論文透過 Web 上大量的資訊作為基底，透過機器學習的方式能即時性知道供應鏈之關係。

本論文透過 Web 上大量的新聞報導、官方網站、相關網頁等內容，擷取出與公司、產品相關之特徵，經由機器學習的方法建立分類模組，利用機器學習的方法取代專業人士所需之經驗，讓沒有經驗的使用者也能輕易上手。

在公司與產品關聯部分以公司、產品和公司與產品作為關鍵字進行資料蒐集，以搜尋結果數、共現(co-occurrence)、NDCG(normalize discount cumulative gain)、餘弦相似度、PMI(pointwise mutual information)和相對熵(KL-divergence)作為特徵。產品與產品關聯部分則是以產品一、產品二和產品一與產品二作為關鍵字，加上自定義的「產品 x 產品關聯性」以及「分類模組可信度」作為新的特徵。我們將關聯問題視為二元分類問題，採用 Support Vector Machine(SVM)作為訓練以及測試之方法。

在實驗部分以精確率(precision)、召回率(recall)、F1-measure 以及準確率作為效能評估之依據。其中公司與產品關聯的準確率為 0.844，產品與產品關聯的準確率為 0.742。

本論文的內容組織如下：第二章描述相關研究。第三章為系統架構，包含資料蒐集與特徵擷取之方法。第四章為實驗數據以及實驗結果。第五章提出結論和未來研究方向。

2. 相關研究

資訊大爆炸時代來臨，網路上的資源不斷增加，搜尋引擎成為全球性的線上資料庫，如何訂定特徵並從大量的網路資源

中挖掘出有用的訊息，即為一大挑戰。Turney 透過計算相關評論為正向或負向建立一個推薦系統[1]。計算每個形容詞或副詞與 excellent 和 poor 的互信息(mutual information)，以此推論出評論的語意傾向，並判斷是否推薦給使用者。利用 410 筆關於汽車、銀行、電影和旅遊地點的評論作為測試資料，其準確率為 74%。高霆耀及莊秀敏利用 Web 上的資訊藉此產生地址與店家的配對[2]。利用 Conditional Random Field 從網頁中大量的 POI(Point of information) 資訊中辨識出地址與商家名稱，再從 Web 上地址與商家的搜尋結果中，透過搜尋結果數、皮爾森相關係數、餘弦相似度等共 27 個特徵去推斷該商家是否位在該地址上。

為了從文字文件中擷取特徵，需要利用自然語言處理技術(National language processing)[4]。史丹佛的辨識系統 Named Entity Recognizer[1]提供自然語言處理中最常使用的兩種功能，斷詞以及詞性標計(part-of-speech tag)。斷詞系統可將句子拆分成各個詞彙。詞性標計則是能標記每個詞彙的類別，像是名詞、動詞、形容詞等等。本論文中只使用斷詞功能。

機器學習包含許多方法，像是 Naïve Bayes、Support Vector Machine(SVM)、Decision tree、Deep learning[2]等等，本論文採用 SVM 作為訓練及測試之方法。

3. 系統架構

本系統包含四個部分：第一部分為資料蒐集，第二部分為特徵擷取，第三部分為訓練及建立模型，最後是分類預測，如圖一所示。給定公司與產品關聯或產品與產品關聯其中一類之配對，從 Web 上蒐集資料，經由特徵擷取模組產生特徵，並透過訓練模組產生模型，或利用分類預測模組預測。

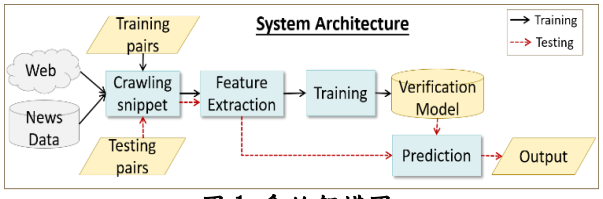


圖 1. 系統架構圖

3.1 資料蒐集

將擁有大量資訊的 google 作為搜尋引擎，利用網頁爬蟲抓取該關鍵字的前十篇搜尋結果以及結果數。每個配對分別產生三種不同的關鍵字，公司與產品關聯的關鍵字分別是公司、產品和公司與產品；產品與產品關聯的關鍵字則是產品一、產品二和產品一與產品二。公司產品配對以及產品一與產品二配對皆由工研院所提供。這項搜尋來源的限制是同 IP 不能頻繁地向 google 搜尋引擎蒐集資料。

3.2 公司與產品關聯特徵擷取

為了有效地辨識出公司和產品關聯，我們定義以下七種共十五個特徵。其中 T_b 代表公司的前十篇搜尋結果，T_p 代表產品的前十篇搜尋結果，T_{b+p} 代表公司與產品的前十篇搜尋結果。

表 1. 公司與產品關聯特徵表

Id	Name	Indicators of a strong relation
1	log *C(b)*	The # of search results for b in log-10 scale
2	log *C(p)*	The # of search results for p in log-10 scale
3	log *C(b+p)*	The # of search results for b+p in log-10 scale
4	R(*b+p*|*b*)	(The # of search results for b+p) / (# of search results for b)
5	R(b+*p*|*p*)	(The # of search results for b+p) / (# of search results for p)
6	P(*b+p*|T_b)	The # of co-occurrence for b+p in T_b snippets
7	P(*b+p*|T_p)	The # of co-occurrence for b+p in T_p snippets
8	P(*b+p*|T_{b+p})	The # of co-occurrence for b+p in T_{b+p} snippets
9	NDCG(p|T_b)	$DCG_p = rel_1 + \sum_{i=2}^{m} \frac{rel_i}{log_2(i)}$, If b occur in the m-th snippet T_p
10	NDCG(b|T_p)	
11	cos(T_b,T_p)	$\frac{\sum_{i=1}^{n} B_i P_i}{\sqrt{\sum_{i=1}^{n} Bi^2}\sqrt{\sum_{i=1}^{n} Pi^2}}$ $B_i \in T_b,\ P_i \in T_p,$

[1] NER, http://nlp.stanford.edu/software/CRF-NER.shtml

[2] ML, https://en.wikipedia.org/wiki/Machine_learning

12	PMI($\bar{b}, p$)	Pointwise mutual information for the relation between business and product which at least one is not zero in T_b, T_p and T_{b+p} p(b)= # of docs with b / $\lvert T_b \cap T_p \cap T_{b+p} \rvert$ $PMI(x,y) = log\frac{p(b,p)}{p(b)p(p)}$
13	PMI($b, \bar{p}$)	
14	PMI(b, p)	
15	KL(D_b, D_p)	The more similar categories for business & product. The KL divergence is computed the divergence of business and product in the category distribution. KL(b, p)= $\sum_{i=1}^{m} b_i \, ln\frac{b_i}{p_i}$ $i \in$ category Select keywords from BoW model as features.

特徵一到特徵三利用搜尋結果數取對數而得。本論文中認為公司或產品的搜尋結果數越低，代表其不存在的機率越高，該配對被分類為沒有關係(False)的機率就會越高。以公司與產品作為關鍵字的搜尋結果數越大，可視為公司與產品間的關聯性越高。特徵四和特徵五透過計算條件機率取得，該特徵數值越大代表關聯性越大。

特徵六到特徵八採用 co-occurrence 的方法計算，利用公司、產品或是公司與產品作為關鍵字的搜尋結果，計算公司和產品同時出現的機率，數值越大代表兩者一起提到的機率越大，關聯性也就愈高。

我們認為若該公司出現在該產品的搜尋結果之第一篇，其關聯性較大；反之若在最後一篇，甚至沒有出現，兩者關聯較小。因此採用 NDCG 作為第九和第十個特徵。NDCG 是種計算排名的方法，常用來測量搜尋引擎的演算法是否有效，其公式如下。

$$DCG_p = rel_1 + \sum_{i=2}^{m} \frac{rel_i}{log_2(i)}$$

第十一個特徵是餘弦相似度，數值越高代表兩者的相似度越高，關聯性也就越高。

每個配對會有三種關鍵字，共三十篇搜尋結果，以這三十篇結果去計算 PMI，即可得到特徵十二到十四，PMI 公式如下。

$$PMI(x,y) = log\frac{p(b,p)}{p(b)p(p)}$$

第十五個特徵是相對熵，利用公司和產品的類別機率分布去計算兩者間的距離，越近表示相關性越高。透過預測某公司為某類別的機率，此機率即為類別機率分布，產品的類別機率分布亦是如此計算。在以下內容中，我們將同個關鍵字的搜尋結果視為同篇文章，因此每篇文章共包含十個搜尋結果。

首先，我們得找出較顯著的詞彙來提升預測的準度。將所有公司的搜尋結果轉為簡體後，利用史丹佛的斷詞系統進行斷詞，並統計所有詞彙作為第一次斷詞集合(W)。計算每個詞彙出現的頻率後，再透過下述兩個條件進行詞語篩選。

(1)$\exists\, j: tf_{ij}^{B} > 3$ (2) $df_i^B > 0.05 \times |B|$

分別代表(1)存在某個公司的文章，使得該詞語出現在該文章中十篇搜尋結果的頻率大於三。(2)該詞語在所有公司的文章中出現之頻率超過百分之五。

表 2. BoW 篩選詞語之符號表

Symbol	Description
B	The union of all business in given-pair set.
T^B	$T^B = \bigcup_{b \in B} T_b$
df_i^B	Doc. frequency in T^B for word *i*
tf_{ij}^B	Term frequency for word *i* in T_j

將篩選過後的斷詞當作特徵，並利用公司所屬之類別，透過 libSVM 訓練針對公司之所有類別的 BoW 模型(所有類別為公司類別以及產品類別之集合)。產品的 BoW 模型亦同上述方法產生。

透過 libSVM 以及 BoW 模型取得該公司和該產品的類別機率分布(如表 3 所示)，將其正規化並透過公式計算出相對熵，即產生第十五個特徵。

KL(b, p)= $\sum_{i=1}^{m} b_i \, ln\frac{b_i}{p_i}$ $i \in$ category

表 3. 類別機率分布表

機率分布	類別 1	類別 2	……	類別 m
公司	b_1	b_2	……	b_m
產品	p_1	p_2	……	p_m

3.3 產品與產品關聯特徵擷取

為了有效地辨識出產品間的關聯，定義以下八種共十六個特徵。其中 T_p 代表產品一的前十篇搜尋結果，T_q 代表產品二的前十篇搜尋結果，T_{p+q} 代表產品一與產品二的前十篇搜尋結果。

前十四個特徵皆同公司與產品部分，故不多加說明，以下會詳細介紹第十五個及第十六個特徵。

表 4. 產品與產品關聯特徵表

15	Product x product	The larger value in product × product matrix
16	Conf. of classifier of BoW	The higher confidence of classifier for the bag-of-word model

- **產品 x 產品關聯性**

由於產品和產品較不容易在同篇搜尋結果中一起提到，故透過產品和公司的關係以及公司和公司的關係去推斷產品間是否有較高之關聯性，若越高則視為兩者為上下游關係之可能性越大。以下介紹該特徵之產生方式。

首先從訓練資料的配對中統計所有產品以及產品所屬之公司。不論關鍵字為何，從所有搜尋結果中統計某產品與某公司同時出現的次數，產生矩陣 V，將其反制後產生矩陣 V^T。同理，統計公司與公司同時被提及之次數，產生矩陣 U。

將 $V^T * U * V$ 即得到產品*產品矩陣。透過輸入的產品一和產品二，取得矩陣中的值作為第十五個特徵，若找不到該值以零取代。

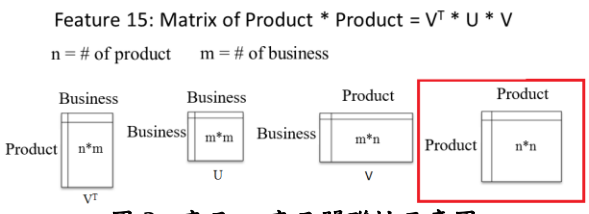


圖 2. 產品 x 產品關聯性示意圖

- **分類模組可信度**

我們利用 BoW 模型預測產品一與產品二為上下游關係之機率，將其機率作為第十六個特徵。

此處選用「產品一與產品二」作為關鍵字之搜尋結果。將搜尋結果轉為簡體後，透過史丹佛的斷詞系統分別對兩類結果進行斷詞，產生 W_T(上下游關係之斷詞集合)和 W_F(平行關係之斷詞集合)。找尋 W_T 與 W_F 之差集，再加上 W_T 與 W_F 聯集中出現頻率大於另外一類兩倍之斷詞，即可得到 $W_T^{(1)}$ 和 $W_F^{(1)}$，規則如下。

$$(1)\ (W_T - W_F) \cup (df_i^{W_T} > 2 \times df_i^{W_F} | i \in W_T \cap W_F)\ \text{as}\ \mathbb{W}_T^{(1)}$$

$$(2)\ (W_F - W_T) \cup (df_i^{W_F} > 2 \times df_i^{W_T} | i \in W_T \cap W_F)\ \text{as}\ \mathbb{W}_F^{(1)}$$

計算 $W_T^{(1)}$ 中每個詞彙在上下游類別中出現的頻率後，利用以下條件進行第二次篩選，並得到 $W_T^{(2)}$。

$$(1) \exists\, j: tf_{ij}^T > 3\ (2)\ df_i^T > 0.05 \times |T|$$

同理對平行關係部分做相同處理，得到 $W_F^{(2)}$。此處與公司與產品關聯部分相同，可參考表 2。

聯集 $W_T^{(2)}$ 與 $W_F^{(2)}$ 產生最終的斷詞集合 W_{T+F}，並以 W_{T+F} 中所有斷詞作為特徵，利用 libSVM 訓練 BoW 模型。針對輸入的產品一與產品二進行分類預測，將其值當作第十六個特徵。

4. 實驗

為了驗證本研究能有效分辨出公司與產品和產品與產品間之關係，實驗分為公司與產品關聯以及產品與產品關聯兩部分，並分別使用不同的資料來源。

4.1 資料集與評量方法

有關聯之公司與產品配對數共 2,114 筆，沒有關聯的有 1,395 筆，合計 3,509 筆。從 google 上抓取的結果有 39,544 篇，加上工研院提供 35,313 篇產業新聞(news)，共 74,857 篇。產品間為上下游關係之配對數為 2,905，平行關係為 2,067，共 4,972 筆。其中 65,770 篇搜尋結果來自 google，65,770 篇來自產業新聞，合計 131,540 篇。以下以 BP 代稱公司與產品關聯，PP 代稱產品與產品關聯。

表 5. 整體實驗數據

Type	Class	Total	Description
BP	T	2,114	39,544 snippets from google
	F	1,395	35,313 snippets from news
PP	T	2,905	65,770 snippets from google
	F	2067	65,770 snippets from news

兩種關聯我們皆視為二元分類問題，故採用 libSVM 作為訓練及測試之方法，並利用精確率、召回率、F1-measure 以及準確率進行效能評估，其算法同一般的計算方式。

在本章節中會顯示三種實驗結果，分別為(1)模型效能 (2)學習曲線 (3)特徵重要度。以下圖中的 news 代表以產業新聞為訓練資料之模型，google 代表資料來自 google，n+g 則代表混合兩種資料。

4.2 公司與產品關聯分類效能

以二比一的比例對公司與產品關聯進行三次交叉驗證，將效能加總平均後得到圖 3 與圖 4 之實驗結果。訓練配對數量為 2340 筆，測試配對數量為 1169 筆。

從圖 3 可以看出，以產業新聞與 google 作為訓練資料的模型之整體效能高於另外兩個模型。

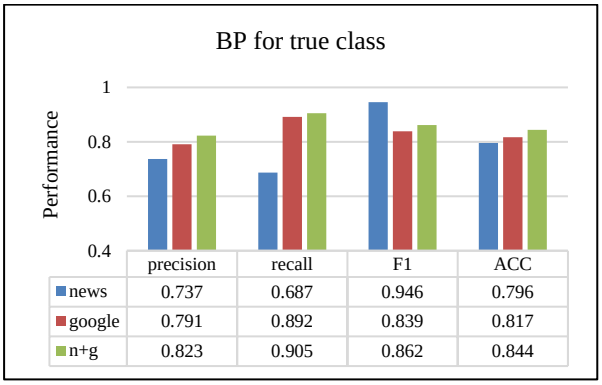


圖 3. Performance of BP true class

True class 與 False class 結果差不多，以產業新聞與 google 作為資料來源之模型效能較佳，F1 從 0.63 提升 33%來到 0.82，準確率也從 0.737 改善到 0.844。

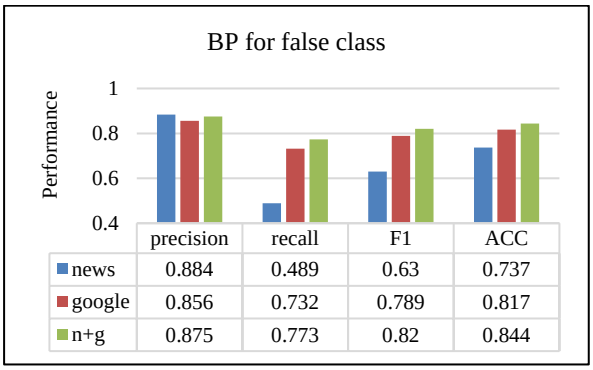


圖 4. Performance of BP false class

圖 5 為學習曲線，可看出訓練資料越大時，效能越好。測試配對數為 509 筆。

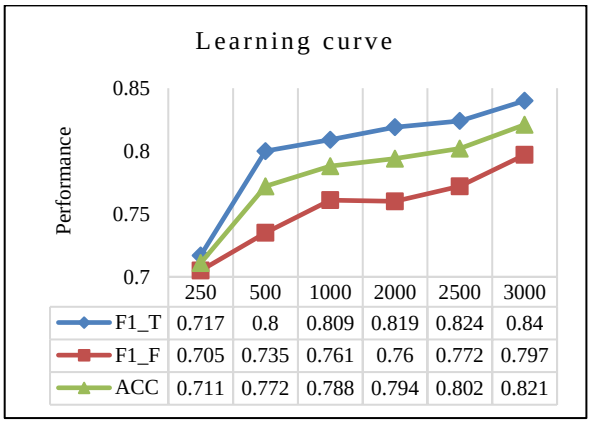


圖 5. Learning curve for BP

從圖 6 可看出有三種特徵是有用的，其中 PMI 為最有影響力之特徵。

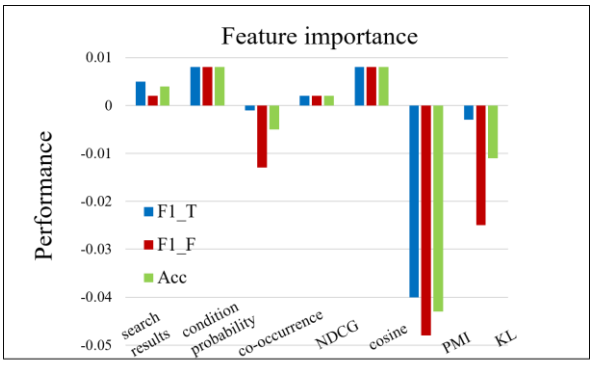


圖 6. Feature importance for BP

4.3 產品與產品關聯分類效能

此部分利用三種不同的資料源作為訓練資料，分別建立模型並進行實驗。訓練配對有 4500 筆，測試配對則有 472 筆。

以產業新聞與 google 作為訓練資料之模型較佳，準確率共提升 13%，如圖七。

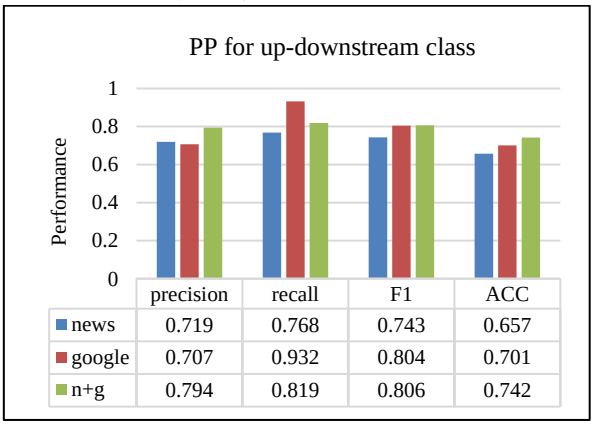
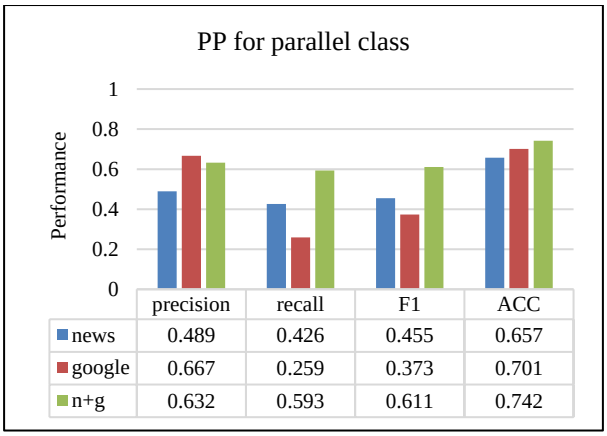


圖 7. Performance of PP up-downstream class

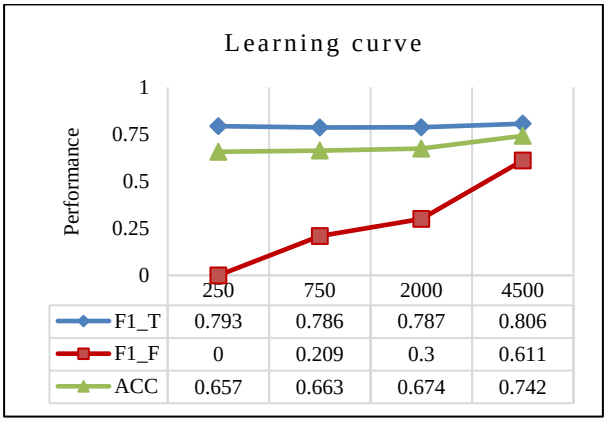
平行關係的實驗結果為圖 8。以產業新聞和 google 作為訓練資料的模型將 F1 從 0.373 改善到 0.611，共提升 64%



	precision	recall	F1	ACC
news	0.489	0.426	0.455	0.657
google	0.667	0.259	0.373	0.701
n+g	0.632	0.593	0.611	0.742

圖 8. Performance of PP parallel class

下圖為產品與產品關聯之學習曲線。從紅線可看出，資料量的增加明顯提升平行關係之 F1(F1_F)，而上下游關係之 F1(F1_T)和準確率亦有小幅上升。



	250	750	2000	4500
F1_T	0.793	0.786	0.787	0.806
F1_F	0	0.209	0.3	0.611
ACC	0.657	0.663	0.674	0.742

圖 9. Learning curve for PP

從圖 10 可看出，絕大多數特徵皆為正影響，其中又以第八種特徵之影響力最為顯著。

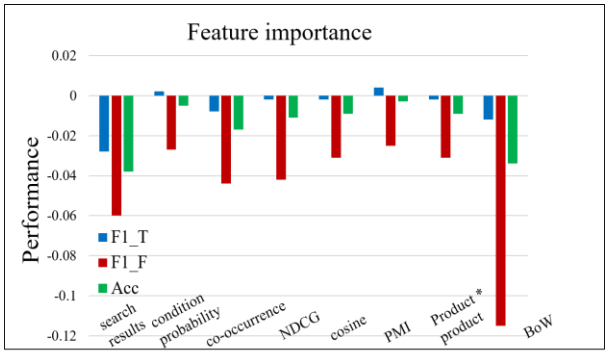


圖 10. Feature importance for PP

5. 結論

在本篇論文中，分別針對公司與產品、產品與產品的關係提出兩種分類驗證模組。從 Web 上進行資料蒐集，從資料中擷取特徵，利用機器學習的訓練方式，取代過去依靠人力吸收的大量知識及經驗，其準確率在公司與產品關聯是 0.844，產品與產品關聯部分為 0.742。從整體效能的比較圖中可看出，以產業新聞與 google 作為訓練資料的效能較佳，由此可知，訓練資料越多樣化，效能就越好。

資料量的多寡也會影響效能，從學習曲線的圖中可看出，公司與產品關聯的訓練資料從 250 增加到 3000，準確率從 0.711 提升到 0.821，共提升 15%；產品與產品關聯的部分雖然準確率從 0.657 提升到 0.701，只有小幅提升 6%，但是針對平行配對的 F1(F1_F)有大幅提升。

學習曲線的明顯成長代表若訓練資料不足夠，效能就會降低。而要取得正確的公司與產品或產品與產品關聯配對並非易事，本論文中的配對關係由工研院所提供。在未來研究方向上，若能從企業或國家取得更大量、更多元的資料，系統效能應能再次提升。此外依然有些許特徵為負相關，尋找其他正相關的特徵，也能改善其效能。

致謝

本研究由工學院計畫部分贊助。

參考資料

[1] TURNEY, Peter D. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews.

[2] In: *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002. p. 417-424.

[3] PANG, Bo; LEE, Lillian; VAITHYANATHAN, Shivakumar. Thumbs up?: sentiment classification using machine learning techniques. In: *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, 2002. p. 79-86.

[4] 高霆耀；莊秀敏；張嘉惠. 基於 Web 之商家景點擷取與資料庫建置. *on Computational Linguistics and Speech Processing ROCLING XXVII (2015)*, 2015, 180.