以樹狀結構及新詞判斷分類XML文件

黃宇翔
成功大學資訊管理研究所

潘柏璇
成功大學資訊管理研究所

摘要

延伸標記語言(eXtensible Mark-up Language; XML) 規格是由全球資訊網標準製定組織(W3C)制定，並於1998年2月成為推薦規格。XML已逐漸成為網路上不同系統和資料庫間資訊交換的新標準，加上其結構化的特性，使得在處理大量XML文件分類成為一個重要課題。目前XML在文件分類上有利用Naïve Bayes演算法、樣版辨識和影像處理分割技術、詞性標記和法則式技術以及TFIDF以解決分類問題等方法，由於過去的研究鮮少針對文件本身的內容作分析，可能造成含糊文件或衍生的相關文件無法正確分類。本研究先以文件的樹狀結構特性找出每個項目的重要性等級，並利用TFIDF方法取得特徵項目後，便可藉由比對各類別的特徵項目將文件正確分類。在分類過程中，同時考量文件中的重要新詞以提高分類正確率。為使分類器能不侷限在限有特徵項目中，本研究也提出一個加入重要特徵項目的機制，使分類器能適應廣泛內容的文件。本研究最後與同樣使用階層特性的XML文件分類方法作一比較，結果顯示本研究能顯著改善分類之正確率。

關鍵字：延伸標記語言、樹狀結構、文件分類、關聯資訊萃取、新詞

An Approach of Classifying XML Documents with Tree-Like Structure and New-Term Usage

Yeu-Shiang Huang
Institute of Information Management, National Cheng Kung University

Po-Hsuan Pan
Institute of Information Management, National Cheng Kung University

Abstract

The extensible mark-up language (XML) devised by the W3C has been a universally accepted and recommended specification. Recently, XML has gradually become a standard information interchange protocol for different systems and databases on the web. In addition, since XML has a characteristic of structural syntax, the classification of the tremendous amount of XML documents is thus of special essential in the field of knowledge management. Various approaches have been proposed on XML classification, such as Naïve Bayes, template reorganization, image processing, tagged analysis, and TFIDF, etc. However, these approaches rarely focused on analyzing contents of documents, and thus sometimes result in incorrect classifications. In this paper, we employed a tree-like structure to obtain the importance of each term, and utilized the TFIDF calculation to attain the special terms in documents. The classification can therefore be process by identifying these special items among documents. The use of new-term in documents is also under consideration to leverage the accuracy of classification. Finally, the proposed approach was compared with other similar approaches, and the results showed that the proposed approach can significantly improve the accuracy of classification.

Key words: XML, tree-like structure, classification, association extraction, new-term

C.E.P.S.

壹、緒論

在現今數位經濟時代，使用者在面臨龐大的資料量時，往往需要耗費許多時間來尋找需要的資料，因此如何將資料有效管理以便能取得重要的資訊就成為一相當重要的課題，而在文件管理的議題中，使用者所直接面臨到的問題就是如何將文件有效分類，如能將文件做好適當的分類與管理，不但能提升組織內部資訊共享的便利性，更能有效掌握外部資訊的衝擊與變化。由於目前資訊取得多以網頁內容為主，但是HTML先天上的限制，使得網頁只具有讓人們觀看的功能，而網頁本身並沒有意義。而XML正能補足HTML在結構上不足之處，讓使用者自行定義有意義的標籤，且讓其他電腦得以辨認，使資料在網路上有自動交換的能力。

半結構化資料模式能表達文字和多媒體文件，而他的浮現使得網路發展和電子化文件能夠被等同看待。其模式可允許將文件內容和邏輯結構編碼，也能透過不同型態的中介資料(meta-data)將文件內容豐富化。過去許多研究認為XML和半結構資料模式有相似之處Goldman et al. (1999)，且XML是具描述性語言的結構化文件，它的發展使得在表達方式增加複雜度和對於文件中不同元件間的差異來開發一個工具成為必要性。其中資訊擷取(Information Retrieval; IR)在處理廣泛文件上有主要的發展，而IR方法應適合這些新型態的文件。由於文件分類(document classification)隸屬於資訊擷取範疇，且對於文字和影像的的分類模式在結構化文件出現之前就已有發展(Denoyer et al. 2003)，基於上述，由於XML已成為網路資訊交換的新標準，加上XML結構化特性，非常適合用在文件分類技術上。過去XML文件分類技術有使用Naïve Bayes演算法，利用條件機率並配合字根還原(stemming)、斷字(stopwords)、門檻值(thresholding)等方法將文件自動分類。在文件辨識上，利用樣版辨識和影像處理分割技術將XML文件中的文字、圖形、影像分割成部分區域和屬性，並依據分類邏輯作分類。詞性標記(Part-Of-Speech tagger) 和法則式(Rule-Based)技術先將文件中某些具有意義或需要訓練的字做上標籤，再放置以法則為基礎的分類器作訓練以得出分類訊息。此分類器能分類各種不同格式的訊息，並使用XML做為文件管理和與其他系統的界面。另外還有使用TFIDF(term frequency / inverse document frequency)方法找出XML文件中的特徵項目，以做為文件分類的依據。

XML 1.0在1998年二月由W3C所推薦，一份有效的XML文件也是一個有效的SGML文件，其已逐漸成為資訊表達和資訊交換的標準(Bray et al. 2000)。最常見的結構化文件格式，如XML和SGML已普遍存在，然而用在資訊擷取中的結構化搜尋卻非常少，Trotman (2004)藉由分析XML語言來提高搜尋辨識能力，一一分析文件樹中的多個節點，並利用TFIDF方法求出每個字在文件中的相關性。在過去十年中，以內容為基礎的文件管理工作(即資訊擷取，information retrieval; IR)在資訊系統領域都佔有顯著的地位，而資訊擷取中常見的分類模式有布林模式、向量模式、機率模式以及類神經網路。Salton et al. (1983)提出布林模式資訊擷取的改進方法，Mihalcea與Moldovan (2000)利用布林資訊擷取系統將字的語義加入到檢索中，以便建立語義檢索(semantic indexing)。在向量模式中，Salton et al. (1975) 將向量空間模式應用在自動檢索，Bernstein et al. (2003)藉由連結

實體(entities)來分類，並利用關連向量空間模式來擷取出整個連結架構，此架構透過向量的權重來表達每一個實體。Liu et al. (2002)改進傳統向量空間模式中字詞權重的公式，並對向量空間模式提出多階層文字分類方法。

機率模式是在1976年開始發展，Fuhr與Pfeifer (1994)認為機率資訊擷取主要有三個概念，抽象、歸納學習和機率假設，作者整合這三個概念，並提出一個新的機率模式。Ng et la. (2001)提出一機率模式並結合本體論(ontology)將Web文件分成相關文件和非相關文件兩類，實驗結果證明作者提出的機率模式在Web文件中的二元分類比向量空間模式(VSM)的正確率還高。類神經網路和分類樹是樣版分類的基本方法，Chen與Chu (1995)結合類神經網路、分類樹和智慧搜尋策略，並發展類神經分類樹模式(NNCT)，此模式中每個節點會包含一個類神經網路，並能提供有效的搜尋演算法，且能在不影響正確率的情況下減少計算複雜度。Bruzzone與Melgani (2003)提出進階分類系統希望能對遠端感應圖像獲得正確且可靠的監督式分類。

TFIDF分類器是以Rocchio (1971)提出的相關性回饋(relevance feedback)演算法為基礎，Aizawa (2000)提出特徵量(feature quantity)的資訊理論觀點，即特徵可以數量化表示，此觀念是基於某一文字和某一文件兩事件同時發生的機率，並加上TFIDF測量方式來計算某一文字的特徵量。Joachims (1996)依據Rocchio演算法中的機率分析發展出PrTFIDF分類器，此分類器由TFIDF衍生而來，並且在文字分類中較TFIDF有更好的表現。關聯是一個很強大的資料分析技術，Wong et al. (1999)對資料採礦中的關連法則提出一改進方法，他能對大量文集資料的內容做文字採礦，並且可使內容的關連視覺化。Berger et al. (2004)在資訊擷取系統中利用關連網路提出一種知識表達模式，作者藉由結合關連網路和強迫性擴散活動方法來構成此一搜尋演算法，並能找出字集的關係。自動部分詞性標記會依據句子中的意思將每個單字配給適當的詞性Meteer et al. (1991)。Brill (1992)提出一個簡單的以法則為基礎的部分詞性標記，此方法可以自動取得法則和標記。Scott與Matwin (1999)透過片語、同義字找出文字的句法和語義關係，而從文件中萃取名詞片語首先將每個字的詞性標記起來，再將標記的字聚集起來形成名詞片語。在XML文件分類應用中，Sung et al. (2002)提出智慧型文件分類方法TARPA，此方法先將文件放到一個Tagged-Regions並擷取較少且重要的部分做分析和分類。Denoyer et al. (2004)提出一個新的統計分類模式能分類結構化文件和多媒體文件。

本研究發現XML文件鮮少針對文件本身的結構化特性來定義出每個項目的重要性作為分類的依據，也很少透過文件本身的內容作語義分析，並進一步將文件中的項目(term)依關聯性作分類。文件的分類依據其內容關聯強度而隸屬於某一類別，有些文件單純性較高，其內容從頭到尾不偏離某一分類主題，此類文件可以用客觀的方法將其直接歸類；然而當有些文件的內容牽涉到的主題不止一類時，或者文件中出現的新興名詞過多時，此時利用客觀技術將此文件分類就有其困難度，否則必須倚賴人為的經驗來給予適當分類。換句話說，由於一份純文字內容的文件主要是由項目(term)所組成，因此當我們要分辨一份文件是屬於哪一個類別時，可以藉由判斷其項目是屬於哪一類別來將此文件分類。但是當一份文件牽涉的內容過於廣泛時，我們往往就很難將此類文件精確的分類到某一特定的主題上。基於此一問題，本研究希望針對此種不易分類的文件，提出一個有效的解決方法。另外本研究認為文件的內容會依社會變遷或科技的發展等環境因素而

不斷的改變，此一現象連帶的就會關係到項目的變化，包括新穎的名詞或者流行詞句等等的出現，因此一個有效的分類器必須能夠因應文件發展和變化的可能性，此一議題也是我們另一個重要的研究動機。

基於上述，本研究希望找出各項目在結構中的重要性，並同時針對一些無法直接依據內容中的項目直接得到分類的文件提出一個解決方法。本研究認為將這類文件其代表性項目和相關類別的關聯強度可以得到正確的分類，因此當某一新的XML文件無法從已知類別的代表性項目中得知屬於它的類別時，希望提出一個機制來解決這個問題。另外本研究也希望提出一個自動新增重要新詞的機制，以便能夠取得新的特徵項目。因此本研究目的有二點：

（一）利用XML文件之結構化特性及對於文件中具重要性的新詞，能透過項目間的關聯加入適當的類別，以強化分類的正確率。

（二）提出一個具有新增特徵項目的分類器，使得廣泛的文件都能夠透過此分類器得到正確的分類。

而透過文件正確率的提升，能增進知識管理之效能，對於企業或組織之經營當有顯著之助益。

貳、以樹狀結構及新詞判斷分類XML文件

當XML文件無法以客觀的方式明確分類時，亦即無法依據文件中各個項目詞句所展現出來的內容將文件分類時，如何將這類文件進一步分析及找出其與可能類別的關聯強度並正確分類是一個重要的課題。

本研究先將XML文件分類，並利用TFIDF、XML語法中的標籤部分，和領域知識的相關字彙三個方法來得出每個分類的特徵項目，最後再將未知文件依上述方法找出代表性項目，並和每個分類的特徵項目一一比對，最後得出XML文件的分類結果。已知分類文件透過斷字處理，以及取得每一訓練文件的特徵項目陣列，可得出每一類別的特徵項目，並且可依重要性大小依序排列，如表2.1所示：

表2.1：訓練文件產生特徵項目

重要性排名 / 分類	特徵項目				
	1	2	3	4	…
A	TermA-1	TermA-2	TermA-3	TermA-4	…
B	TermB-1	TermB-2	TermB-3	TermB-4	…
C	TermC-1	TermC-2	TermC-3	TermC-4	…
·	·	·	·	·	…
·	·	·	·	·	…

在A、B、C等類別中分別可以找出每一類別的特徵項目，在A類別中，項目A-1的重要性最高，其次為項目A-2，以此類推，而B、C等其他分類亦同。同樣的，未知分類的

XML文件的項目在經過處理後也能得出每一份測試文件的代表性項目，並將這些代表性項目依重要性等級排列。一般我們會將測試文件的代表性項目與類別中的特徵項目作比對，當所有已知類別的代表性項目與類別中的特徵項目比對完成後，便可得知每個類別的權重值，並將此份文件歸類到得分最高的類別中。

上述的過程假設文件中所有項目的重要性都是相同的，然而依據目前文件表達習慣，一個項目出現在文件標題與文件內容時，所表示的重要性並不相同，此時給予相同分數並不公平。且多數文件內容往往會包含相近幾個類別的特徵項目，若忽略每個代表性項目重要性的差異，可能因反客為主而發生分類錯誤的情況。若只以現有已知類別的代表性項目將文件分類，有可能某些對文件具重要性的潛在類別，因其所屬的重要項目為新詞，而使得該類別對文件的重要性因為沒有這些重要新詞給予支持，導致此潛在類別被判定為不具重要性，而無法成為文件之類別。隨著知識的演進，同屬一個類別的文件所探討的重點可能有所改變，此時我們將無法使用原先的特徵項目進行新XML文件的分類，因此一個好的分類器應能夠判斷重要的新詞並將之作為特徵項目，才能適應廣泛的文件。因此本研究提出一個新的XML文件分類的方法，來改善含糊文件無法直接分類的問題，並提出一個具有學習機制的分類器，其運作流程如圖2.1所示：

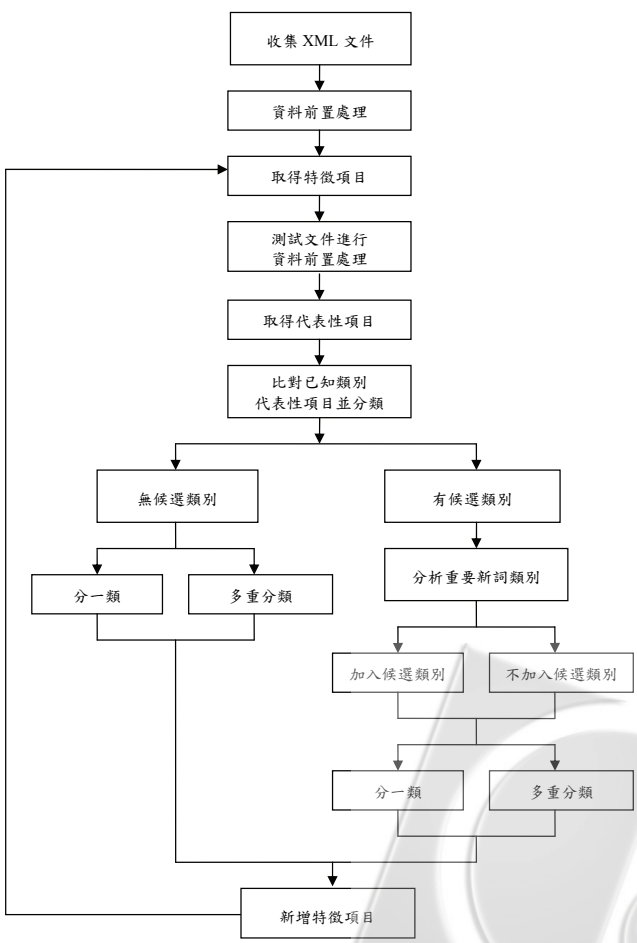

圖2.1：XML文件分類方法流程

本研究提出的XML文件分類方法之重要步驟，包括「收集XML文件」、「資料前置處理」、「取得特徵項目」、「測試文件進行資料前置處理」、「比對已知類別代表性項目並分類」、「分析重要新詞類別」、「測試文件分類」、「計算正確率」以及「新增特徵項目」，分別探討如下：

（一）收集XML文件：本研究首先會從網頁上蒐集XML文件，將所有文件分類之後，取其中一部份作為訓練文件，另一部份作為測試文件用。

（二）資料前置處理：在訓練文件之前，本研究會先將XML的每個字依據其不同型態，和在文件中不同的階層，給予不同的重要性分數。

（三）取得特徵項目：本研究先分析文件中標籤所擁有的重要性等級，再利用TFIDF計算每個單字的權重，從已分類文件中找出各類別的特徵項目。

（四）測試文件進行資料前置處理：本研究將測試文件，並依照上述取得特徵項目的方式找出每一份文件的代表性項目。

（五）比對已知類別代表性項目並分類：一份測試文件在取得代表性項目後，便會與類別中的特徵項目作比對，若比對到則在該類別上給予計分，直到得出每個類別對此份文件的權重值為止。

（六）分析重要新詞類別：測試文件經TF運算後會得知每個類別對文件的重要性，有時可能會出現潛在的重要類別，就必須要藉由未知類別的重要新詞來判斷這些候選類別是否為文件的分類結果，本研究利用在文件中與重要新詞相關的已知類別字詞來判斷新詞的類別，便可判斷最後的分類結果。

（七）測試文件分類：新詞處理主要目的是為決定候選類別是否可成為文件的分類類別，若新詞不屬於候選類別的文字時，則候選類別無法晉級為文件的類別，若候選類別的權重在加入新詞的同屬類別而成為文件的類別，但在分類之初，也可能有一個以上的類別通過分類門檻值而為多重分類，或者只分到一個類別中；有可能文件在開始之初沒有被分到任一類別中，而最後只歸為一類；也有可能因納入候選類別而為多重分類。

（八）計算正確率：分類完成後，本研究會判斷每個Fold測試文件的分類結果是否為原來收集文件的類別，並計算分類正確率。

（九）新增特徵項目：不同測試文件依序分類後，會判斷出每個測試文件的新詞類別，並在累進的文件中得到新詞的TFIDF值，直到新詞對某一類別的重要性達到該類別最低特徵字門檻值時，表示此一新詞足以成為此類別的特徵項目，應予以新增進來，作為往後文件分類判斷之依據。

參、XML文件分類方法

一、由文件的樹狀結構求得權重

XML文件可讓使用者自訂標籤，也可以設定屬性，其語法為<標籤名稱 屬性名稱=“設定值”>，以圖3.1為例，文件的主體部分是由成對的標籤所組成，而標籤與標籤之

間的文字則為字元資料，即<標籤>字元資料</標籤>，字元資料是用來敘述元素的內容文字。最上層的標籤稱為根元素，例如<spam-document></spam-document>為開始標籤與結束標籤，而spam-document則為元素。

```
<codeColoring>
    <scheme name="ASPes no" id="ASP_JScript" priority="20">
        <ignoreCase>No</ignoreCase>
        <ignoreTags>Yes</ignoreTags>
        <keywords name="Reserved Keywords"
id="CodeColor_ASPJSReserved">
            <keyword>break</keyword>
            <keyword>case</keyword>
        </keywords>
    </scheme>
</codeColoring>
```

圖3.1：XML文件範例

Denoyer與Gallinari (2004)提出XML文件中，透過標籤之間的階層關係，在標籤裡的所有屬性具有相同的重要性，並藉由屬性中的設定值來解釋屬性的特性，以明訂XML文件的完整性。以scheme標籤來說，屬性name 、id 、priority具有和scheme同等的重要性，而"ASPes no"、"ASP_JScript"和"20"只是為解釋屬性所給的特定值，其重要性便次於前二者。在權重值設定上，Jenkins與Inman (2000)提出的網頁文件自動分類方法中，作者依據每個字在HTML階層的分佈位置，給予不同的分數。出現在<TITLE>的文字給予兩倍的分數，出現在<H1>標籤的文字給予一倍分數，在<H2>標籤的文字則只佔一半分數。根據上述作者以二的指數設定分數之方式，本研究先以每個字在樹狀結構的分佈情形標示階層數字，根元素及其屬性為1，而屬性值和字元資料為2，第二層標籤裡的元素和屬性為2，屬性值和元素的字元資料為3，以此類推。因此本研究將第一層元素和屬性的權重值設為1，而屬性值和字元資料的權重值為1/2，以此類推，即可將XML文件中的每個字給予適當的權重值，其計算公式如下：

$$W_i = \frac{1}{2^{R_i - 1}} \qquad R_i = 1, 2, 3, 4, \ldots \quad \cdots\cdots (3.1)$$

其中W_i為第i個字的權重值，R_i為每個字給予的等級，以圖3.1之XML文件為例，根元素為codeColoring，第二層標籤的元素為scheme，以此類推，經等級排列大小後可得到每個字的重要性分數如表3.1所示：

表3.1：XML文件中各單字的重要性分數之範例

i	Term	R_i	i	Term	R_i
1	codeColoring	1	13	name	3
2	scheme	2	14	id	3
3	name	2	15	No	4
4	id	2	16	Yes	4
5	priority	2	17	Reserved	4
6	ASPes	3	18	Keywords	4
7	no	3	19	CodeColor_ASPJSReserved	4
8	ASP_JScript	3	20	keyword	4
9	20	3	21	keyword	4
10	ignoreCase	3	22	break	5
11	ignoreTags	3	23	case	5
12	keywords	3			

二、類別特徵字及分類

藉由上述方法，可以得出所有已知類別XML文件每個字的權重值，在計算出屬於各類別的字及其權重後，本研究會將所有相同類別的字和權重值放在同一文字檔中，因此文字檔的個數即是所要分類的類別個數。之後便將各類別文字檔裡的文字做字根還原和斷字處理，以得出最後具有意義的文字。字根還原是為將文字中的單複數名詞等不同詞性所造成的文字差異加以還原，以得到原始名詞的字。字根還原後，再對照斷字表，將不具有意義的文字刪除，最後便能初步留下可作為各類別分類的特徵字庫。

為找出能代表各類別的特徵字，Salton et al. (1983)提出之TFIDF為$tw_{ij} = tf_{ij}\log_2(\frac{N}{n_j})$，其中$tw_{ij}$為某一文字$k_j$在文件$i$的權重值，$tf_{ij}$是文字$k_j$在文件$i$出現的頻率，$N$是所有文件的個數，$n_j$是含有$k_j$文字的文件篇數。本研究依據上述式子，並配合XML文件的特性，以求出各類別中文字的權重。其計算方式為：

$$tw_{ij} = tl_{ij}\log_2(\frac{C}{c_j}) \quad \cdots\cdots (3.2)$$

其中tw_{ij}為某一文字k_j在類別i中經過TFIDF運算後的權重值，tl_{ij}是上述中，文字k_j在類別i因不同階層所得到的權重值之加總，C是所有類別的個數，c_j是在所有類別中，含有k_j文字的類別個數。本研究tw_{ij}之運算為將某一類別之文件全部匯總，以求得其總權重值。將所有經過TFIDF運算後的總權重值予以排列後，各類別分別取前10%作為類別的特徵字，之後便可將測試文件進行分類，由於本研究主要是將文件中的元素、屬性、屬性值和字元資料依據其在階層分佈的特性給予不同的重要性，再藉由TFIDF的運算後，各類別中前10%的字應有足夠的條件作為類別特徵字，實務上亦可對此比率作調整，以檢視其對正確率之影響。測試文件會依據上述方法找出文件中每個字的權重值，並在排列後將權重值最大的前15%的代表性項目作為判斷此測試文件要分類的字庫，此15%之選擇則是因應之前類別特徵值之選取，使其稍大於10%，實務上亦可做更動。測試文件在進行分類過程中，會將字庫中的每一個字比對各類別的特徵字，比對到的字會歸屬於那個

類別，而未比對到的字則不予分類。為得知文件屬於哪個類別，本研究將同屬一類的文字，根據該字在測試文件中的權重值做加總，以得出各類別在此份文件的權重排名，例如若同屬A類別的文字權重經加總後的得分最高，則此測試文件就屬於A類。

上述例子是當A類別在此份文件的重要性明顯高於其他類別時，則此份文件便能很明確歸屬A類，但是若某份測試文件中，所有已知類別的字的權重值經過加總後，發現在各類別中，屬於A、B兩類的權重值不相上下，則將此文件歸於此二類。例如當我們在yahoo (www.yahoo.com)的分類搜尋引擎裡尋找 “multimedia”，關於“多媒體”的相關文章就被分類在十個類別，因此本研究認為多重分類在文件分類是一必要的考量要素，若文件只限歸屬於單一類別，則分類搜尋之尋得性(Recall)可能會降低。簡而言之，文件的分類標準取決於各類別對於此文件的重要性，因此高於分類門檻值者，即可視為文件的類別，其判定標準為：

$$T_{class} = W_{maxclass} - (W_{maxclass} - W_{avgclass}) / \frac{n_{class}}{2} \quad \cdots\cdots (3.3)$$

其中$W_{maxclass}$為權重最大值，是所有已知類別的文字中，同屬一類的文字之權重值經加總後的最大值，$W_{maxclass}$為權重平均值，是所有已知類別字的權重值加總除上所有已知類別的類別個數，n_{class}為類別個數，因此利用$(W_{maxclass} - W_{avgclass}) / \frac{n_{class}}{2}$可找出各類別在測試文件中的最大權重值和各類別權重平均值的單一差距，而T_{class}為第一個區間值的分類標準。例如某一測試文件的代表字在比對各類別特徵字，且同類別權重值經加總後分別為A類權重值0.8、B類0.75、C類0.6、D類0.4、E類0.3，則$(W_{maxclass} - W_{avgclass}) / \frac{n_{class}}{2}$ = (0.8-0.57)/2.5=0.092，而T_{class}為0.708，因此此文件會分類到A、B二類。當兩個以上的類別對於此文件的權重值皆大於分類門檻值時，表示這些類別對於文件具有一定的重要性，因此就可將文件判定為多重分類。

三、新詞處理

本研究考慮到有些類別對於此文件之重要性雖未到達分類門檻值，但其重要程度可能在加入文件中重要的新詞後而提升，進而成為文件的分類類別。基於此，本研究先以候選門檻值找出潛在的重要類別，再判斷文件中重要的新詞之類別，最後便可決定潛在類別對文件的影響程度，並將文件做適當分類。候選類別門檻值之計算方式為：

$$T_{cad} = W_{maxclass} - 2((W_{maxclass} - W_{avgclass}) / \frac{n_{class}}{2}) \quad \cdots\cdots (3.4)$$

其中T_{cad}為候選類別門檻值，是以兩個區間作為候選類別標準，當文件中有候選類別時，本研究首先會判定文件中的重要新詞，其判斷方式為：

$$Term_{new} = (W_{maxterm} + W_{minterm})/2 \quad \cdots\cdots (3.5)$$

其中$W_{maxterm}$為最大項目權重值，是文件的所有字詞的權重中最大者，而$W_{maxterm}$為最小項目權重值，是文件的所有字詞的權重中最小者，$Term_{new}$為本研究設定之門檻值，凡文字權重大於$Term_{new}$即是本研究要分析的新詞。

依據Denoyer與Gallinari (2003)，XML文件中的元素具有階層關係，每一子階層的項目都是依據父階層的項目所延伸而來，且每一字元資料項目則是依據他本身階層的項目，而文章的內容可藉由句子中名詞與動詞之間的關聯得到項目間的關聯資訊，因此本研究認為，透過分析文件中，新詞所屬的元素、屬性和屬性值，以及字元資料的文字內容中各名詞所屬的類別，便可將新詞分類到適當的類別中。在判斷未知新詞的類別時，本研究首先找出未知新詞在原始文件中的所在位置，再藉由分析未知新詞所屬的階層元素的類別，以及與未知新詞相關之屬性和屬性值的類別，來判斷未知新詞的類別。當所有重要的未知新詞都得知其所屬類別後，我們便將與候選類別具有相同類別的新詞之權重值累加，若累加後的候選類別權重值有超過分類門檻值時，則此類別便能成為文件的分類類別。

在新增各類別特徵字的過程中，只要文件中有出現重要新詞，先判定這些新詞的類別，並依據不同類別放入各個字庫中，因此只要有新進名詞，都會重新將字庫的項目和權重作更新，並計算每個新詞的TFIDF值，因此若某一新詞有出現在其他類別，則它的重要性對此類別而言就會顯著降低。在計算完各新詞的TFIDF值後，判斷是否有達到進入類別特徵字的門檻值，門檻值標準為各類別經排列後的前10%字庫中，最後一個特徵字的權重值，因此若某一新詞的權重值有達到前10%字庫的最低門檻值，即可成為此類別的特徵字。新增特徵字的方式是以判斷後進文件的重要新詞為主要依據，我們無法以一篇文件就斷言重要的新詞是否能成為類別的特徵字，因此需不斷分析後續的文件來定位新詞對類別的意義，但是在評定的過程中，也可能會把一個常出現的通用字納入為特徵項目，為避免判定錯誤的情形發生，本研究仍會持續追蹤已加入特徵字的新詞，只要從往後的文件中判定新詞為不重要的特徵字時，便會將此字從詞庫中刪除。

肆、實證研究

本研究依據Yahoo!入口網站(http://www.yahoo.com)對商業類網站的分類類別，從NewsGator網站(http://www.newsgator.com/)分別針對Economics、Finance、Investment、Marketing和Trade等五個類別各蒐集45篇文件，並編號為1至45，各類別的文章篇數如表4.1所示。

表4.1：本研究之各分類項目名稱及XML文件數目

類別	分類項目名稱	XML文件數目
A	Economics	45
B	Finance	45
C	Investment	45
D	Marketing	45
E	Trade	45

在訓練文件之前，必須將XML的每個字依據其在文件中不同的階層，將標籤中的元素、屬性、屬性值及標籤外字元資料的字給予不同的重要性。依照上述方法，根元素在文件的重要性分數為1，第二層標籤裡的元素和屬性為2，屬性值和元素的字元資料為3，以此類推，便可得到文件中所有單字的重要性等級，表4.2為某篇訓練文件的每個項目依據階層和結構型態給予重要性等級的部分實例。

表4.2：訓練文件中各項目及其重要性等級

i	Term	R_i
1	RDF	1
2	admin	1
3	channel	2
4	creator	3
5	generatorAgent	3
6	resource	3
7	license	3
8	economics	4
9	catches	4
10	eye	4
·	·	·
·	·	·

為使每篇文章能確認分類的正確率，將各類的45篇文件等分切割成九個Fold，並以亂數方式產生1至45個數字，隨機選取五篇文章放入各個Fold中，並依次將每一個Fold的五篇文件作為測試文件，其餘八個Fold作為訓練文件。本研究依據每個字在文件中的分佈位置和型態給予不同重要性分數。依據方程式(3.1)，每個字的權重值依其重要性分數有所不同，分數1的權重值為1，分數2的權重值為0.5，分數3為0.25，以此類推。並利用(3.2)式子得出每個類別的特徵項目。為能以有效率的方式將文件正確分類，將各類別排列後的特徵字，取前10%作為文件分類的特徵字。

從訓練文件中得到每一類別的特徵字後，就可將測試文件作分類，測試文件如同上述方法，可以依文字型態和不同階層得到每個項目的重要性分數，並將每個字依(3.1)式計算權重值，最後前15%具重要的代表性項目，比對各類別前10%的重要特徵項目，來作為分類的準則。比對完成後，本研究將各個相同類別的代表性項目之權重值分別作加總，計算完每個類別對此份文件的權重值後，此文件便可歸類在權重值最高的類別中。本研究同時考量到當有些類別具有某種程度的相關性時，一份文件並無法明確的被歸類在某一特定類別，因此本研究提出一個多重分類的機制，以期將XML文件能分類在適當且相關的類別中。表4.3為某篇測試文件前15%的代表性項目及其權重值，而此測試文件中已知類別的代表性項目與各類別特徵項目經比對後所得出每個類別對此文件的權重值如表4.4所示，表中右邊A至E分別為Economics、Finance、Investment、Marketing、Trade等

五個類別的代號，帶入(3.3)式子計算後可知：=1.40625-(1.40625-1.34375)/1.5=1.36458，因此只有A類別達到此文件的類別標準。接下來便要進一步判斷是否B及E類別的重要性亦足以作為此文件的分類類別。

表4.3：測試文件前15%的代表性項目

代表性項目	權重值	類別	代表性項目	權重值	類別
item	2	-	width	0.0625	B,E
title	1.1875	-	height	0.0625	B,E
link	1.1875	-	economics	0.0625	A
com	0.65625	-	index	0.0625	-
http	0.625	-	cfm	0.0625	-
newsisfree	0.625	A, B, E	syndic	0.0625	-
www	0.59375	-	php	0.0625	-
rss	0.5	-	personal	0.0625	-
version	0.5	-	non	0.0625	-
iclick	0.46875	A, B, E	commercial	0.0625	-
channel	0.25	-	only	0.0625	-
description	0.1875	-	contact	0.0625	A
language	0.125	-	sate	0.0625	-
webmaster	0.125	-	apr	0.0625	-
lastbuilddate	0.125	-	rates	0.0625	-
image	0.125	-	cgi	0.0625	A, B, E
economist	0.125	-	stock	0.0625	-
finance	0.125	-	exchanges	0.0625	-
america	0.09375	-	buttonwood	0.0625	A
url	0.0625	-	column	0.0625	-
			tax	0.0625	A, B

表4.4：各類別在測試文件中佔有的權限值

類別	權重值
A	1.40625
B	1.34375
C	0
D	0
E	1.28125

表4.4中最大權重值之代表性項目item原本因為比對到特徵項目而為此文件的重要新詞，但由圖4.1之此篇測試文件的部分原始內容，發現沒有與item相關的代表性項目，使得無法利用相關代表性項目之類別來推斷新詞類別，因此item之權重值2並無法累加在任何類別中。

```
<item>
    <title>America's stock exchanges</title>
    <link>http://www.newsisfree.com/iclick/i,81775607,1677,f/</link>
</item>
<item>
    <title>American International Group</title>
    <link>http://www.newsisfree.com/iclick/i,81775606,1677,f/</link>
</item>
<item>
    <title>Banking in South Africa</title>
    <link>http://www.newsisfree.com/iclick/i,81775605,1677,f/</link>
</item>
```

圖4.1：測試文件之部分原始內容

雖然B及E類別未到達文件分類的標準，但仍須考量到測試文件中同屬此兩類別的重要新詞在加入所屬類別的權重後成為分類類別的可能性，因此本研究會先判斷文件中是否有候選類別，再進一步分析文件中的重要新詞，以作為最後的分類結果。帶入(3.4)公式得知 $T_{cad} = W_{maxclass} - 2((W_{maxclass} - W_{avgclass}) / \frac{n_{class}}{2})$=1.40625–2×0.04167=1.32291，當達到門檻值時，即表示此文件有候選類別，便要判斷文件中重要新詞的類別。根據表4.3之代表性項目，並依(3.5)公式之 $Term_{new} = (W_{\max term} + W_{\min term})/2$=(2+0.0625)/2=1.03125，得知此篇文件的重要新詞為item, titile和link三個項目，再找出和此三個項目在原始文件中相關代表性項目，並分析其類別。相關代表性項目的選取方式為取出在相同標籤中，含有重要新詞的所有項目，但由於item在原始文件中並無相關代表性項目，因此無法判定新詞item之類別。由圖4.1的重要新詞title為例，其第一個標籤裡的相關代表性項目有america, stock及 exchanges，因此本研究會進一步分析這些項目的類別，如果有比對到各類別的特徵項目，則會在該類別加上特徵項目的權重值，直到計算完相關代表性項目在各類別的權重值為止。利用上述方法後判定titile和link此兩重要新詞的類別皆為A類別，其權重值皆為1.1875，而候選類別仍無達到分類門檻值，且A類別的權重值在加入titile和link的權重值後，A類別對此測試文件的權重值便提升為3.78125。

為實驗本研究在權重值的設計上是否能提升分類準確度，本研究依序使用9 個Fold來測試各文件分類的結果，並對分類結果作一研究分析。表4.5為分類結果之正確率。

表4.5：分類結果之正確率

Fold	正確率	Fold	正確率	Fold	正確率
1	92%	4	80%	7	92%
2	80%	5	88%	8	84%
3	92%	6	80%	9	96%
				Avg.	87.11%

由表4.5可發現本研究分類方法之正確率皆能維持在80%水平之上，其中第九個Fold的測試結果更高達95%以上的正確率，而Fold2, Fold4, Fold6, Fold8也顯示某些訓練及測試文件會造正確率較低。本研究認為除了標籤裡的項目依據其階層結構特性有不同重要性之外，一般字元項目亦會依據其在不同階層的分佈位置有不同的重要等級。本研究整合階層特徵項目和一般特徵項目的重要性等級，並將文件中所有特徵項目取前10%作為類別特徵字，此方法能同時兼顧各種文字型態的重要性，使得文件分類時能客觀地依據不同型態的特徵字提高分類正確率。

除了透過分析各不同型態特徵字權重分數以提高正確率以外，本研究進一步提出了分析文件中重要新詞，使分類結果更為完善。本研究發現在測試第三個Fold的某篇B類別的文件中，在比對完已知類別的代表性項目後的各類別分數如表4.6所示，經公式(3.3)、(3.4)計算後得知達到分類門檻值為E類別，但由於A及B類別為候選類別，在經過分析重要新詞的類別並累加在該類別分數後的結果如表4.7所示，由表中可知此文件不但正確的被歸類在原來所屬的B類別，也由分析後得知此文件為歸屬於A、B及E類別的多重分類。

表4.6 各類別在測試文件中佔有的權重值

類別	權重值	類別	權重值
A	0.3125	D	0.0625
B	0.3125	E	0.375
C	0.1875		

表4.7：測試文件分類結果

類別	權重值
A	1.75
B	1.0625
E	0.375

本研究以分類門檻值作為文件分類的準則，使得多重分類為分類結果帶來不同的可能性，以表4.7的分類結果為例，此篇金融類的文章亦涵蓋了經濟和貿易的相關內容。本研究利用各類別在測試文件中的最大權重值和各類別權重平均值的單一差距來找出分類門檻值，並以第一個區間值作為分類的標準，以下一區間值為候選類別標準。以表4.7為例，當最大權重值0.375扣除單位區間門檻值0.05，表示只要有達到0.325的分類標準的類別對文件都有相當的重要性，而將最大權重值0.375扣除兩個單位區間門檻值0.05時，即可進一步分析較次等的候選類別。單位區間門檻值可依實務需要調整。圖4.2為各類別特徵項目個數及其分類正確筆數之關係：

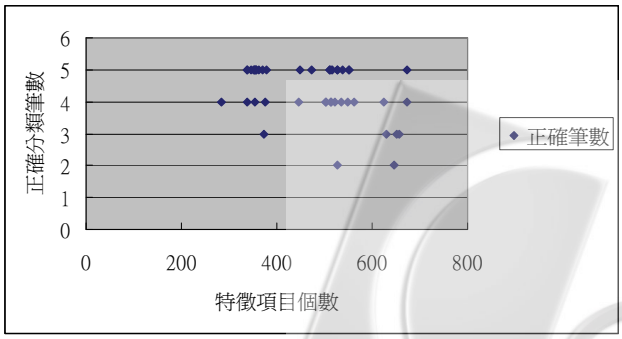

圖4.2：各類別特徵項目個數及其分類正確筆數之關係

由圖4.2中可知類別的特徵項目個數與分類結果並無絕對關係，有些特徵項目雖只有三百多個，卻能使分類正確率高達100%，而某些類別的特徵項目個數雖高達六百多個，卻沒有增加分類的正確筆數，本研究發現正確筆數為2和3的類別多為E類別Trade，因此若只就正確筆數4或5及其特徵項目之關係來看，三百至四百的特徵項目個數都能維持在80%以上的正確率，因此本研究取前10%的比例作為特徵項目個數，可為文件分類提供適當的特徵字，且不會造成過渡訓練的現象。本研究分別將九個作測試的Fold中，各類別及其對應的正確筆數以圖4.3所示：

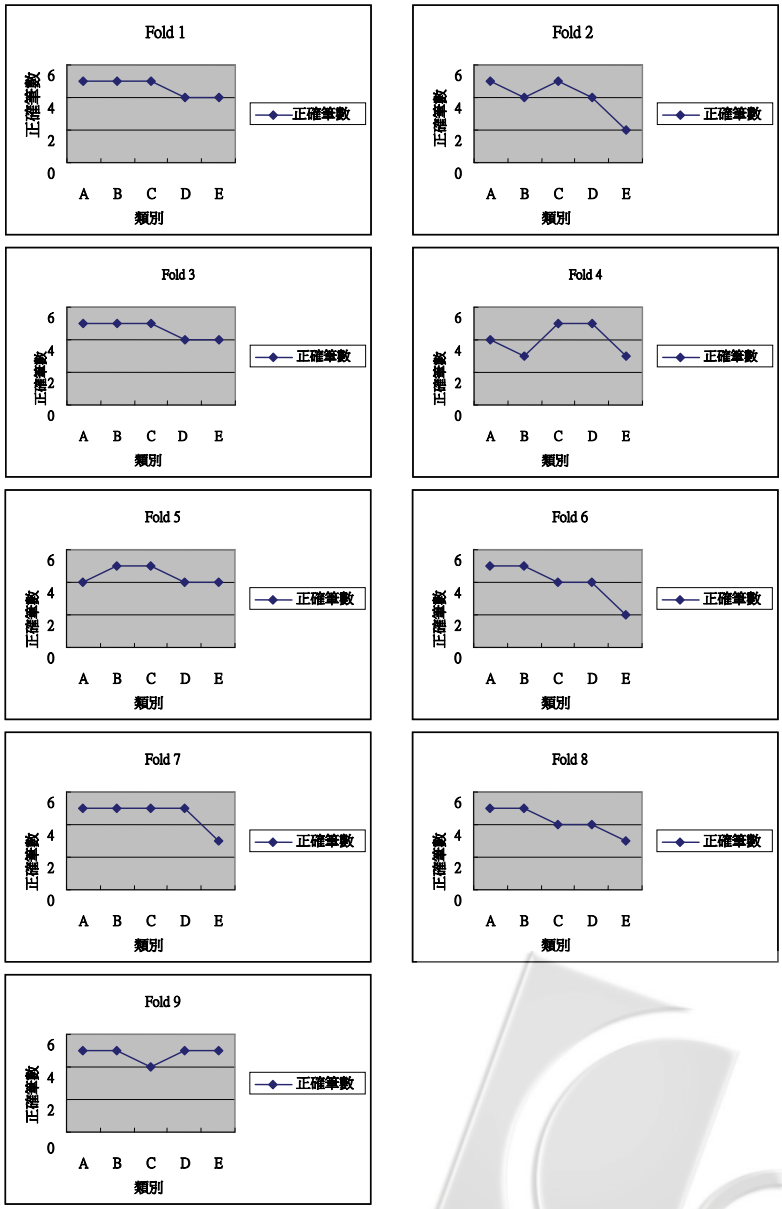

圖4.3：每個Fold中各類別分類的正確筆數

本研究對測試文件是依據權重值高低將前15%的代表性項目比對各類別前10%的特徵字，原本在實驗的過程中，採取和特徵項目相同比例的10%作為代表性項目的個數，但實驗結果並不如預期的理想，原因是因為在收集的XML文件中，有許多標籤內的元素項目多為通俗的字，雖然出現的頻率非常高，但是因為它對每個類別而言並不是有意義的項目，反而一般字元資料多能表達類別的項目。由圖4.3可知，A到D個類別的正確率多能維持在80%以上，而E類別的正確筆數多分佈在2至4的個數中，經本研究對貿易類的分類結果分析後發現，除了該類別以外，貿易類文件的分類結果多被歸類在經濟和行銷的類別，本研究認為此種結果可能是因為有關經貿交流、貿易行銷或經貿投資等相關文章皆與不同類別有直接或間接關係，使得分類結果廣泛地在各個不同類別中。為探討本研究提出判斷重要新詞類別方法之正確率，將每個作測試的Fold中，不同類別產生新詞之類別的正確率列於表4.8，正確率是依據各Fold中，A至E類的測試文件所判斷的重要新詞之類別是否為該類別。

表4.8：不同Fold中判斷重要新詞類別之正確率

Fold	類別					
	A	B	C	D	E	Avg.
1	92%	76.9%	73.9%	72.7%	18.75%	66.85%
2	84.6%	76.2%	95.2%	81.25%	60%	79.45%
3	100%	78.6%	100%	72.2%	73.3%	84.82%
4	64.7%	56.5%	95%	84%	47.4%	69.52%
5	75%	87.5%	92.6%	71.88%	73.9%	80.176%
6	100%	78.6%	75%	62.5%	22.7%	67.76%
7	73.1%	66.67%	65%	79.2%	33.33%	63.46%
8	92.3%	86.2%	60%	91.67	76.47%	81.328%
9	100%	70.8%	81.5%	60%	91.67%	80.794%
Avg.	86.85%	75.33%	82.02%	75.04%	55.28%	

由表4.8可知除了E類別的重要新詞以外，各Fold中，A至D類測試文件重要新詞之類別有75%以上能正確判斷在該類別，但正確率的變動幅度很大，並不能很穩定維持在同一水平中，顯示本研究提出的判斷新詞之類別方法可以補足文件在分類之正確率，但在E類別中，可能因誤判新詞之類別而影響正確率。表4.9為本研究分別將各類別文件取出不同比例作為訓練文件及測試文件，以驗證分類正確率在縮減訓練文件數的情況下是否能維持不變。

表4.9：不同比例訓練文件及測試文件分類正確率之比較

類別	訓練文件(80%) 測試文件(20%)	訓練文件(60%) 測試文件(40%)
A	88%	83%
B	100%	100%
C	88%	88%
D	100%	94%
E	66%	61%
Avg.	88.4%	85.2%

在測試20%的文件中，除了類別A和類別C有一個錯誤筆數外，類別B和類別D皆高達100%的正確率，但E類別的分類正確率仍偏低。另外當只有以60%的訓練文件取得特徵項目時，雖然會些微降低分類正確率，但還是維持在80%以上的分類水平。為驗證重要新詞對分類的正向影響程度，本研究將兩種訓練文件和測試文件判斷重要新詞類別結果之正確率列於表4.10：

表4.10：判斷重要新詞類別之正確率

類別	訓練文件(80%) 測試文件(20%)	訓練文件(60%) 測試文件(40%)
A	86.8%	85.5%
B	90%	73.33%
C	91.3%	80.3%
D	80.56%	77.5%
E	12.12%	13.8%

由表4.10可知，當訓練文件有充分篇數萃取出類別特徵項目時，測試文件中的重要新詞較能夠依據適當且足夠數量的特徵字來判定出正確的類別；但是當減少訓練文件篇數，新詞在比對較貧乏的特徵項目時，容易導致錯誤分類的結果。表4.11為以第三個Fold作測試文件後，判斷為D類別的重要新詞及該新詞在測試文件中的權重值。

表4.11：以第三個Fold的測試文件產生D類別之重要新詞

權重值	重要新詞項目
5.9375	title
4	description
3.75	item
2.8125	subject
2.65625	com
2.3125	line
2.125	create
2.125	marketing
2	link
1.28125	channel
0.78125	http

E.P.S.

依據表4.11，本研究發現大多數的新詞為原始文件中標籤內的元素，這些字出現的頻率過高，使得它在文件中佔有很高的權重值，在依序測試每一篇文件後，本研究會根據存在字庫中的新詞，重新計算這些新詞的TFIDF值，只要TFIDF值有達到該類別特徵字的最低門檻值，此新詞就會加入成為特徵項目，而D類別特徵字在測試完25篇文件的新增結果如表4.12所示：

表4.12：新增D類別特徵項目

TFIDF	特徵項目
0.273583756	thread
0.273583756	kinds
0.272559412	website
0.266502536	weblog
0.26344539	person
0.26344539	talking
0.26344539	learn
0.26344539	team
0.26344539	competition
1.103342902	line
1.013882666	create
1.013882666	marketing
0.611311608	channel
0.495256666	subject

對照表4.11及表4.12可以發現，雖然title, description, item等新詞具有非常高的權重值，但由於這些字亦有出現在其他類別的新詞字庫中，使得其TFIDF值無法達到特徵字的門檻值，而不會將這些字作為特徵項目，因此從表4.12中可發現，只有少數具有意義的新詞被納入特徵字庫中。

伍、結論與建議

XML文件藉由標籤與標籤之間的樹狀階層關係構成具有意義的結構化文件，本研究利用XML的結構特性，將文件中各不同型態的文字給予不同的重要等級，本研究認為一份文件在分類的過程中，不應只考慮已知類別的代表性項目，而提出新詞處理的機制，實驗結果證明了本研究以XML文件之結構特性，並輔以加入重要新詞權重值的方法，能顯著提升文件分類之正確率。再者，為考量文件內容的廣泛及多樣性，本研究將提出一個能自動新增特徵項目的機制，此分類器能自動將新穎的特徵項目自動加入，使XML文件不致造成無法分類的窘境。

本研究希望藉由XML文件結構化的特性發展一個能自動新增特徵項目的分類器，因此只以XML文件作為研究對象。再者，只針對單字詞(single-term)做為文件內容研究的對

象，並不考慮名詞片語或多字詞(multi-term)在處理上的問題，最後本研究不考慮中文文件的文字處理方式，只以英文內容作為實驗資料。目前有關XML文件分類的研究多以英文文件為主要資料，因此未來的研究可朝向以漢字為主的文件分類上的文字分析處理方法。另外可將文件的階層特性結合其他不同的分類方法，以探討不同分類方法對XML文件的有效性。在評斷新詞類別中，未來可針對不同文字處理方法來探討利用其他方法找出新詞的類別。最後可針對所有特徵項目作全面更新動作，使得分類器能自動篩選出重要的特徵字以提升分類正確率。

參考文獻

1. Aizawa, A. "The Feature Quantity: An Information Theoretic Perspective of TFIDF-like Measures," *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2000, pp. 104 – 111.
2. Brill, E. "A Simple Rule-Based Part of Speech Tagger," *Proceedings of the Third Conference on Applied Natural Language*, ACL, Trento, Italy, 1992, pp. 152-155.
3. Berger, H., Dittenbach, M., and Merkl, D. "An Adaptive Information Retrieval System Based on Associative Networks," *Proceedings of the First Asian-Pacific Conference on Conceptual Modeling* (31), 2004, pp. 27-36.
4. Bernstein, A., Provost, F., and Clearwater, S. "The Relational Vector-space Model and Industry Classification," *Working Notes of the IJCAI-2003 Workshop on Learning Statistical Models from Relational Data* (SRL-2003), L. Getoor and D. Jensen, editors, Acapulco, Mexico, 2003, pp. 8-18.
5. Bray, T., Paoli, J., Sperberg-McQueen, C. M., and Maler, *E. Extensible Markup Language (XML) 1.0*, 2nd edn., W3C Recommendation, Technical Report REC-xml-20001006, World Wide Web Consortium, 2000.
6. Bruzzone, L., and Melgani, F. "An Advanced Classification System Based on the Back-propagation of Consensus," *IEEE International Geoscience and Remote Sensing Symposium* (IGARS'03), 2003, pp. 1785-1787.
7. Chen, Y. S., and Chu, T. H. "A Neural Network Classification Tree," *IEEE International Conference on Neural Networks*, (1), 1995, pp. 409-413.
8. Denoyer, L., and Gallinari P. "Bayesian Network Model for Semi-Structured Document Classification," *Information Processing and Management*, 2004, pp. 807-827.
9. Denoyer, L., Vittaut, J. N., Gallinari, P., Brunessaux, S., and Brunessaux, S. "Structured Multimedia Document Classification," *Proceedings of the ACM Symposium on Document Engineering*, 2003, pp. 153-160.
10. Fuhr, N., and Pfeifer, U. "Probabilistic Information Retrieval as a Combination of Abstraction, Inductive Learning, and Probabilistic Assumptions," *ACM Transactions on*

Information Systems (TOIS) (12:1), 1994, pp. 92-115.

11. Goldman, R., McHugh, J., and Widom, J. "From Semistructured Data to XML: Migrating the Lore Data Model and Query Language," *Proceeding of the 2nd International Workshop on the Web and Databases*, 1999, pp. 25-30.
12. Jenkins, C., and Inman, D. "Adaptive Automatic Classification on the Web," *11th International Workshop on Database and Expert Systems Application*, 2000, pp. 504-511.
13. Joachims, T. "A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization," *Proceedings of the Fourteenth International Conference on Machine Learning*, 1996, pp. 143-151.
14. Liu, S., Dong, M., Zhang, H., and Shi, Z. "An Approach of Multi-hierarchy Text Classification," *Journal of Chinese Information* (16:3), 2002, pp. 95-100.
15. Meteer, M., Schwartz, R., and Weischedel, R. "Empirical Studies in Part of Speech Labelling," *Proceedings of the DARPA Speech and Natural Language Workshop*, Morgan Kaufmann, 1991.
16. Mihalcea, R., and Moldovan, D. "Semantic Indexing Using Wordnet Senses," *Proceedings of ACL 2000 Workshop on Recent Advances in NLP and IR*, 2000.
17. Ng, Y. K., Tang, J., and Goodrich, M. "A Binary-Categorization Approach for Classifying Multiple-Record Web Documents Using Application Ontologies and a Probabilistic Model," *Proceedings of the 7th International Conference on Database Systems for Advanced Applications (DASFAA 2001)*, Hong Kong, 2001, pp. 58-65.
18. Rocchio, J. J. "Relevance Feedback in Information Retrieval," *The SMART Retrieval System-Experiments in Automatic Document Processing*, ed. G. Salton, Englewood Cliffs, New Jersey, 1971, pp. 313-323.
19. Salton, G., Fox, E. A., Buckley, C., and Voorhees, E. M. "Boolean Query Formulation with Relevance Feedback," *Communications of the ACM* (26), 1983.
20. Salton, G., Wong, A., and Yang, C. S. "A Vector Space Model for Automatic Indexing," *Communications of the ACM* (18:11), 1975, pp. 613-620.
21. Scott, S., and Matwin, S. "Feature Engineering for Text Classification," *Proceedings of the Sixteenth International Conference on Machine Learning*, 1999, pp. 379-388.
22. Sung, L. C., Chen, M. C., and Kuo, C. H. "Web Document Classification Based on Tagged-Region Progressive Analysis," Working Paper, 2002 (available on line at http://ranger.uta.edu/~alp/ix/readings/webDocClassification.pdf).
23. Trotman, A. "Searching Structured Documents," *Information Processing and Management* (40), 2004, pp. 619-632.
24. Wong, P. C., Whitney, P., and Thomas, J. "Visualizing Association Rules for Text Mining," *Proceedings of the 1999 IEEE Symposium on Information Visualization*, San Francisco, CA, 1999, pp. 120-123.