以網頁識別及清理改善跨網站資料擷取之研究

劉仁宇
國立中央大學資訊工程所
92532020@cc.ncu.edu.tw

張嘉惠
國立中央大學資訊工程所
chia@csie.ncu.edu.tw

摘要

在網際網路的使用中，資料的收集與整合往往占據相當大的時間，這其中包括了兩個相當煩瑣的工作，首先是網頁的收集，再來是資料的擷取。本篇論文即是以此兩個題目為主軸的研究。我們以國際會議網站為例，希望藉由分類方式，辨識出網頁中屬於論文接受的網頁，再加以擷取其中所公佈的作者及論文題目。首先在接受論文網頁的辨識上，我們提出若干特徵做為 SVM 分類器的依據；而在資訊擷取上則先經過網頁清理再由 Softmealy 完成資訊擷取；其背後的緣由是網頁中經常夾雜著許多的雜訊，藉由網頁清理可以有效的減少擷取的錯誤。實驗數據從 DBWorld 中各會議網站，辨識接受論文所在的網頁，再經由頁面清理擷取主要內容，其結果顯示有相當程度改善效果，也證明頁面清理想法的可行性。

關鍵詞：資料擷取、機器學習

1. 緒論

隨著 Internet 技術的快速發展，在網際網路上以 HTML 格式為主的文件資訊，以直觀與豐富的表達能力帶給使用者獲取新知來源的極大便利，然而這些文件即便是相同領域若來自不同網站，搜尋出的內容，其設計表達方法卻不盡相同，帶來了資料統整上的困擾與挑戰。因而後續開始有了資訊擷取(information extraction) 方法論的研究，以期自動擷取網頁所包含的重要內容或使用者感興趣的事實。

過去資訊擷取系統的研究主要著重在具有樣版格式網站的資訊擷取，對於多樣性網站的資訊擷取及整合問題較少著墨，仍有相當大的改進空間。一般來說，一份文件的內容不僅只有語法結構，還包含了表現樣式、語意或是內文中的其他結構。對於多樣性網頁的資訊擷取而言，雖然各個網頁表現格式烱異，但是語意及內文的其他結構仍有共通特點，以為人所辨識。半結構化文件或是有同樣樣版網頁的資訊擷取多採用網頁的格式做為資訊擷取規則的特性，而自由文體的擷取則多仰賴語言的構詞、語法、語意來做分析，其擷取規則主要是建立在文法關係的基礎上。

Web 文件大多以 HTML 語法撰寫，網頁內文在語義了解上不及自然語言，在格式結構的表達上卻又不及資料庫。因此網頁內文擷取的正確性是一項挑戰。本研究以會議網站為主題，目標在擷取各會議網站中所接受的的文章主題、作者姓名。以圖 1 為例，四個子視窗分別呈現四個國際會議網站的論文接受網頁。會議網站的位址係由 DBWorld 取得，利用此網頁內的連結，下載所有的網頁，再辨識其中公佈的接受論文網頁。所收集的網頁再以 Softmealy[2][3]訓練資訊擷取的規則，由於網頁內容煩雜，我們嘗試以網頁清理方式，找出其中包含接受論文的區塊。綜合言之，本研究架構共分為三階段：

圖 1 樣式多變的網頁結構

1. 接受論文的分類：任一網站下的全部網頁，數量繁多，若由人力去挑選網頁，不僅費時且煩瑣，因此本研究希望能自動分析網站，將人力的介入減至最低，而能自動分類出包含接受論文的網頁以供後續文章標題及作者的擷取。
2. 頁面清理(Page Cleaning)：分類出的 Accepted Papers 中，其網頁內容煩雜，因此需要做網頁清理的工作，找出其中包含接受論文的區塊。利用資料特性的分析技巧，例：DOM Tree 區塊偵測、分行段落偵測及過濾特徵符號等技巧來達成網頁清理的目標。
3. 最後，應用 SoftMealy 擷取器：經由 label 標註，規則學習及 FST 擷取器，達成資料擷取的工作。

2. 接受論文的分類

一般而言，網站中的接受論文的網頁通常存放在此區域範圍內的某個目錄中，而不會鏈結至其他 domain name，因此我們在下載網頁時，也可以忽略其他domain name。公佈接受論文的網頁通常有幾項特性：重覆的資料結構以便呈現多篇論文，以及眾

多的作者名字等，但是要直接找出這些的特性，並不容易。比較容易著手的還是基本的字詞統計，我們統計了訓練資料中屬於接受論文的網頁，針對內文(content)、鏈結文字(anchor text)及鏈結檔案名稱(link filename)依據TFxIDF，列出關鍵字集。

- content_keyword 的關鍵字集：共十一個字詞

  {accept, paper, program, programm, final, research, session, schedul, present, submit, submiss }

- anchor_name 的關鍵字集：共十個字詞

  {accept, technic, research, full, paper, program, programm, schedul, submit, present}

- file_name 的關鍵字集：共十五個字詞

  {accept, acceptedpap, accpap, research, techprog, paper, acc, paperlist, fulllist, detail, program, programm, submiss, present, draft}

除了常見的字詞，接受論文的網頁中由於包含多篇論文，因此常使用表格(table)標籤、列表(list)清單的標籤或是版面規劃(layout)標籤，以表 1 為分類方式，我們分別統計這四類標籤於網頁中出現的個數。同時我們也計算每個網頁文件物件模型樹(DOM tree)的最大扇出數(fan out)。再者是一些常用的標點符號，如冒號、逗號、句號、括號等等在接受論文網頁的分佈或許與非接受論文網頁的分佈有較大的差別，也是我們拿來做為特徵的對象。

表 1 HTML 文件結構使用到的特徵標籤

分類	描述	標籤
Table	表格標籤	table, tbody, tr, th, td…
List	列表清單標籤	ol, ul, li, dl, dt, dd, dir...
Layout	版面規劃標籤	q, br, hr, center, p, pre…
Others	其他標籤	html, div, span, a, em, b, i, u, s, font, h1~h6…

最後我們也使用一些現有工具如詞性標示(POS Tagger)工具及實體擷取(Named Entity Extraction)工具 ANNIE 作為人名的辨別方法[8]。ANNIE 是針對自由文體標示識別存在的各項物件個體的擷取系統，擷取規則的運作機制在於使用 Java 語言開發的 JAPE 文法剖析器，使用正規文法表示式，並結合 LHS／RHS 做遞迴加強規則推導與演繹，加上資料字典的輔助下，幫助使用者從目前環境中擷取出如地名，人名，公司，時間，日期等多項實體。

此三大類特性構成了三個特徵模型，以 Gate's ANNIE 標示的特徵，分別以擴展(Sparse)及壓縮(Compress)兩種模式做為支持向量機(Support Vector Machine, SVM)訓練資料。在 Sparse 模式下每個字詞視為一個特徵，而在 Compress 模式下則取每個字集中的最重要的字做為代表。而另一個模式，則是不使用 NER 工具，直接利用符號特徵、最大扇出樹及 POS 詞性來分析網頁特性。因此在網站分類的實驗下，我們會有三個模型(Sparse, Compress, Org)來做整體的效能評估，完整的特徵如表 2, 3 及 4 所示。

表 2 會議網站特徵(Sparse)

序	特徵欄位	特徵說明	類別
1-11	Content word set	關鍵字	11 個
12-21	Anchor word set	關鍵字	10 個
22-36	File word set	關鍵字	15 個
37	#annie_person	ANNIE 系統標示人名數	

表 3 會議網站特徵(Compress)

序	特徵欄位	特徵欄位說明
1	anchor_name	錨標名稱(若有多個字詞, 以 priority 最高的優先)
2	#anchor_name	anchor_name 字詞個數
3	file_name	連結網址檔案名稱
4	#file_name	file_name 字詞個數
5	content_keyword	內文關鍵字集
6	#content_keyword	內文字詞個數
7	annie_person_count	ANNIE 系統標示人名數

表 4 會議網站特徵(Org)

序	特徵欄位	特徵欄位說明
1-6	anchor_name	同 Compress
7	max_fan_out	最大的扇出個數
8	Table (T)	表格標籤個數
9	List (L)	列表清單標籤個數
10	Layout (Y)	版面規劃標籤個數
11	Others (X)	其他標籤個數
12	AvgTag	最大標籤數/最大扇出數
13-22	POS	POS 系統標示詞性
23	AvgPOS	最大 pos 詞性數/最大扇出數
24	,	逗號個數
25	and / &	字詞"and"或"&"的個數
26	.	句點"."的個數
27	:	冒號":"的個數
28	by	字詞"by"的個數
29	-	特殊符號"-"的個數
30	(	特殊符號"("的個數
31	)	特殊符號")"的個數

3. 頁面清理(Page Cleaning)

第一階段識別分類出的 Accepted Paper 網頁可能因 HTML 網頁標籤雜訊，影響擷取效果。例如，議程的時程說明、議題類別以及中場休息等等，在

此我們藉由網頁清理，將一些無關緊要的標籤做轉換。頁面清理主要針對本文區塊做段行偵測，再對每一個所得到的段落或行標示其是否為雜訊，最後藉由 SVM 訓練雜訊辨識的模型來清理網頁。

本研究採用 Swing HTML Parser 及 JTidy 工具，建構成 DOM Tree 物件模型。實作上以類似 Embley[4]等人提出的最大扇出樹的觀念，即文件中，具有最大扇出數所形成的子樹集合當作主要文字區域。再依序深度優先搜尋（Depth-First Search, DFS）做分行段落的斷行偵測。將追蹤到的<tag>或文字內容，逐項寫入檔案，當巡迴到葉節點時（即為文字內容),若文字節點為 Null 時，不予以寫入；若為控制字元，例：換行符號"\n "，將以空白取代，代表該行內文與上一行文字相關。當搜尋到文字節點下的文字內容時，此文字節點至下一個節點間視為段落邊界(圖 2)。

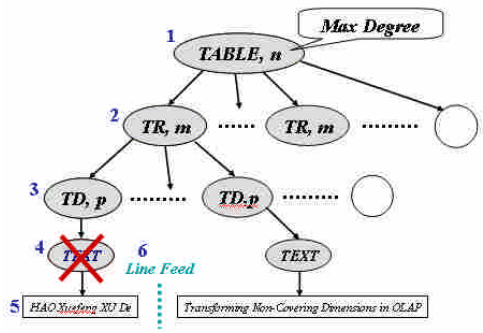

圖 2 區塊追蹤與分行處理示意圖

經過上述的段行偵測，取得分行資訊，加以表 5 所列的特徵屬性，經過人工標示後，做為雜訊過濾的訓練資料。特徵屬性包括前一階段辨識接受論文網頁使用的 Content keywords，是否包含 session、地點、時間、及中場休息等字詞，做為訓練 SVM 過濾雜訊的依據，進而產生雜訊分類 Model。

除了表 5 所列之資料，我們另外以擴展資料格式(表 6)，做為訓練 SVM 過濾雜訊的 Model。兩者效能將於實驗中比較。

表 5 頁面清理特徵(Compress)

序	特徵欄位	特徵欄位說明
1	content_key	是否含內文關鍵字集
2	content_key_no	內文字集字詞數
3	session_part	句中含：session %d%s | session %s%d | session: | session...
4	position	hall, room, hotel, floor, location, location:
5	time_section	morning, afternoon, evening
6	long_rest	lunch, restaurant, rest, breakfast
7	short_rest	coffee break, break, coffee
8-14	同表 6	38-44

表 6 頁面清理特徵(Sparse)

序	特徵欄位	特徵說明	個數
1-20	content_key	關鍵字	20 個
21	session_part	含 session	1 個
22-27	position	hall, room, hotel, floor, location, location:	6 個
28-30	time_section	morning, afternoon, evening	3 個
31-34	long_rest	lunch, restaurant, rest, breakfast	4 個
35-37	short_rest	coffee break, break, coffee	3 個
38	length	句子長度	
39	order_number	含有 1st, 2nd, 3rd, 4th…	
40	numeric	含數值型態	
41	date	含日期型態，例 ：mm-dd-yy | mm/dd/yyyy | token'dd...	
42	time	含時間型態，例 ：am 8:30 | 08:30-12:00 | 8 -12 | pm	
43	year_en	含英文星期的日期型態 例 ：week, month date, year...	
44	week_en	含日期型態 例 ：month date, year...	

4. 應用 SoftMealy 擷取器

經過清理的網頁最後由 SoftMealy 做資訊擷取程式的產生（屬性標示/學習與擷取）的來源網頁。SoftMealy 由許鈞南博士團隊發展的，該項擷取工具是利用 FST 轉換函數的有限狀態機(圖 3)，來模擬語法結構達成半結構化網頁的擷取工作。SoftMealy 提供使用者介面來對範例網頁做標示動作，依據左右位置的標記，推導出 left context 與 right context，再經 generalization algorithm 的 climbing 理論涵蓋歸納出屬性的共通擷取規則。

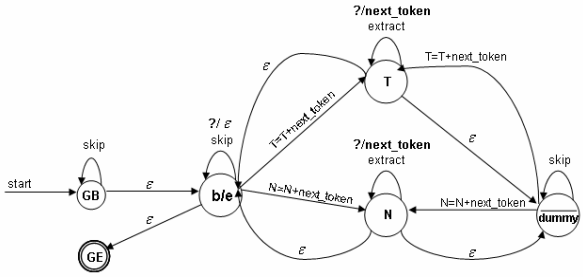

圖 3 SoftMealy 擷取狀態圖

本研究採用 Single-pass SoftMealy Extractor，經由標示、規則學習，進而資料擷取輸出，可從會議網站的接受投稿論文中擷取出投稿文章名稱及作者兩種屬性資料。在後續實驗中發現網頁複雜度會影響 SoftMealy 擷取的正確性，因我們將網頁依所含有論文題目、作者以外的內文、都列為網頁複雜度的因素。

表 7 網頁複雜度定義

複雜度	複雜度說明
1	單純 Accepted Paper
2	單純 Accepted Paper，每筆記錄或 欄位屬性含有未明確邊界的標籤
3	Accepted Paper 雜訊含量較少，每筆記錄 或欄位屬性含有未明確邊界的標籤
4	Accepted Paper 雜訊含量較多，且每筆記錄 或欄位屬性含有未明確邊界的標籤
5	Accepted Paper 雜訊含量多且或欄位屬性含 有未明確邊界的標籤或議程，時間等資訊

5. 實驗

本研究的實驗分為三部份：接受論文辨識、頁面清理、以及以 SoftMealy 擷取原始網頁與清理後頁面兩者間的效能比較。

本研究採用支持向量機(SVM)[5][6][7]，做為網站中的投稿網頁分類器及頁面清理過濾器。SVM 係來自於最佳化理論，擁有高泛化能力，可在高維空間藉由邊界極大化的原理學習到最佳分類超平面，因此具有極佳的分類效果。當樣本數不平衡或是 SVM 反例的數量非常多時，在測試時會產生較大的誤差。主要原因係 SVM 對不同類別的錯誤採用相同的懲罰係數；因此，為了使間隔區間愈大的同時又儘可能的希望藉由 VC 理論以降低風險，分類超平面適必會向樣本數量小的的類別做移動，此舉，將會對小樣本類別產生較大的訓練誤差與測試誤差。有鑑於此，為選擇適當的懲罰係數以提高網站的分類效果，本研究採用林智仁老師 LIBSVM[1] 提供的參數調整工具 grid.py。其主要原理是將訓練數據不斷的疊代與交叉測試所得出的建議最佳參數值。經由實際測試，調整參數 cost 及 gamma 值，確實對網站分類有顯著的改善。

5.1 接受論文辨識實驗

實驗資料來自 DBWorld 所公佈的會議網站，共 55 個網站，基本上每個網站均含有一個接受論文的網頁，少數的網站則是尚未有接受論文網頁的公佈，或是有 2 個接受論文網頁。訓練資料與測試資料如表 8 所示，共計 558 份網頁，其中含 54 份接受論文網頁。

測試集模型的評估準則，採用監督式學習模型評估方法中的錯差矩陣（Confusion Matrix）四個準則來評估分類效果。當反例過多時，錯差矩陣評估準則中的正確率（Accuracy）就愈不重要。錯差矩陣的四項評估準則(表 9)。

表 8 實驗資料採用共 55 個網站共計 558 份網頁

ACM (01-05)	ADC (03-05)	AGENTS (99-01)	APWEB (04-05)
CIKM (02)	COLT (01)	DASFAA (03-04)	DILS (06)
ECAI (06)	ECML (04)	EDBT (00, 02, 04)	ER (98,00)
ICDE (04-05)	ICML (05)	IEEE (06)	IFIP (06)
IJCAI (05)	KDD (01-04)	MDM (00)	P2PIM (06)
PAKDD (01)	SETN (06)	SIGIR (99, 02-04)	VLDB (98-99, 02, 05)
WAIM (06)	WEBDB (98, 00, 02)	WIDM (03-04)	WISE (03-04)
網站：55 個，網頁：558 份，正例：54 份，反例：504 份			

表 9 監督式學習模型（錯差矩陣）

錯差矩陣 （Confusion Matrix）		Predicted	
		Negative	Positive
Actual	Negative	a	b
	Positive	c	d

(1) 精確率：$\text{Precision (P)} = \frac{d}{b+d}$

(2) 涵蓋率：$\text{Recall (R)} = \frac{d}{c+d}$

(3) 正確率：$\text{Accuracy (AC)} = \frac{a+d}{a+b+c+d}$

(4) $\text{F-Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$

一般分類是以高正確率為目標，但當反例過多正例過少時我們則必須以 precision 和 recall 做考量，因此通常以 F-measure 做為評估準則。

■ ***學習曲線評估：***

我們首先做學習曲線評估，先隨機任選一個網站做為訓練資料，其他網站做為測試資料，計算其 F-measure；接著依序增加訓練網站個數，並以剩餘網站做為測試資料，求得分類效果。如此重複數次所得之學習曲線如圖 4 所示。

圖 4 中為三個模型的學習曲線分佈。在訓練樣本數不斷增加時，數據有明顯的曲線上升。採用 Org 模型，在不藉由 NER 工具的標示下，分類效果並不顯著，F-Measure 始終維持在五成五左右的準度；若採用 Compress 壓縮模型，以最重要的特徵字詞來代表，其 F-Measure 約為六成九的準度；最後利用字詞位置的 Sparse 擴展模型，其 F-Measure 有七成五以上的準度。從上述實驗中，三者網站分類模型以 Sparse 為佳。

從曲線上，我們可以觀察出，當樣本數逐漸累加至第六份網站時，Sparse 的變化曲線逐漸趨於和緩，當訓練樣本持續增至第七網站時而達到穩定，因此可判別出此訓練一個可接受的模型約七個網站樣本，即可得出一個不錯的分類訓練模型。

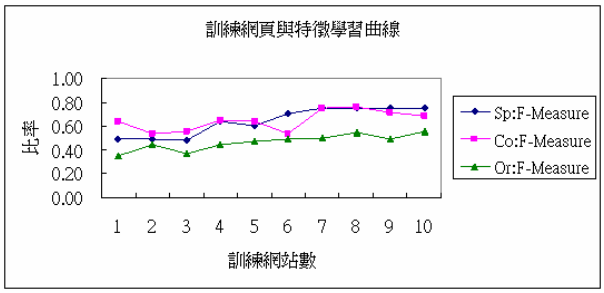

圖 4 訓練網頁與特徵學習曲線

■ ***訓練特徵模型(Compress, Sparse, Org)***

以網站為基準的分類評估，由表 10 三種特徵模型比較來看，整體表現仍屬 Sparse 模型表現較為出色，其 Precision、Recall 及 F-Measure 超過七成的比率，顯示出採用擴展空間加上 Annie 的標註下，我們的實驗模型特徵選取，確實能將會議網站內的論文網頁給辨識出來，且勝過用特徵符號、POS 及最大扇出樹的輔助。同時在實驗過程中也發現核函數的選擇，對分類效果也會產生影響。經由核函數選擇線性或 RBF 以及參數的調整後，其 Precision, Recall, F-Measure 都有明顯的提昇。

因網站下的其他網頁，絕大部份都是反例，其正例的比率非常的低，在 Sparse 模型的回覆率有八成的準度，可將接受論文被 SVM 分類，減少使用者因機器無法識別，而需重新 Review 該網站中的所有網頁。對期望具有高回覆率的需求上，能滿足其需要。

表 10 網站分類效果

特徵模型	Precision	Recall	F-measure	Accuracy
Sparse	0.70	0.80	0.75	0.95
Compress	0.73	0.65	0.69	0.94
Org	0.47	0.68	0.55	0.89

5.2 頁面清理實驗

■ ***學習曲線評估：***

在第二部份的實驗，希望藉由第一階段分類出的接受論文網頁，將使用者不感興趣的記錄（即雜訊）給排除。本研究透過十二組訓練集逐漸遞增所做的測試網頁雜訊過濾結果，實驗數據如下：

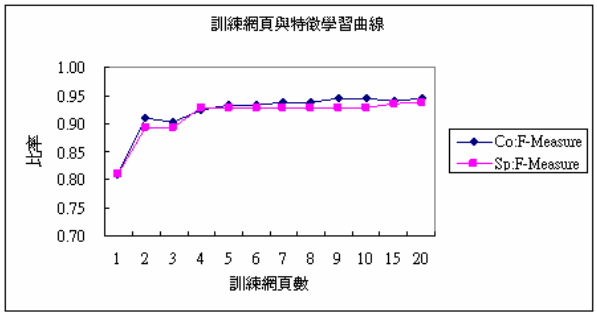

圖 5 訓練網頁與特徵學習曲線圖

我們採用兩種 Compress, Sparse 特徵模型(圖 5)，依不同的訓練樣本，逐漸增加來觀察其模型的訓練曲線，發現第一份網頁當作訓練樣本時，其 Recall 高達 100%，但 Precision 只有 68%；當樣本數不斷增加時，Precision 有明顯的曲線上升，Recall、F-Measure 則是逐漸下降；從兩份至八份的網站訓練結果，有小幅的曲線波動，而當逐漸累加至九份網站訓練樣本時時，訓練曲線逐漸趨於平緩，當訓練樣本增至十份以後的網站而達到穩定，即使給予更多的訓練網站直至廿份網站樣本，其 F-Measure 效果依舊維持九份的準度，因此可判別此訓練一個可接受的模型約九至十份網站樣本，即可得出一個不錯的頁面清理的分類模型。

■ ***訓練特徵模型(Compress, Sparse)***

SVM 在小樣本中會有不錯的效果，過多且不適合的訓練資料集，將會造成過度訓練（overfit）現象，訓練樣本的選擇，勢必會影響分類的效果。經實驗證明，小數量資料集，就可有不錯的分類效果。當訓練網頁增加時，有時會降低其正確率，最主要的原因在於頁面清理的特徵值中有偵測到句子的長度而造成錯誤。因為有可能一篇論文的投稿作者可能僅有一位，此時該作者記錄的句子長度等於 2，長度小於等於 2，一般來說在 Accepted Paper 中絕大部份都被 SVM 分類器歸納屬於雜訊的特徵，因此很有可能被誤判成雜訊而遭到過濾，造成擷取失敗。若要成功的處理此種情況，必須再加上自然語言的處理技術（背景知識）來判斷純文字所代表的意義，才能真正的提高分類效果。

表 11 頁面清理效果

特徵模型	Precision	Recall	F-measure	Accuracy
Sparse	0.89	0.99	0.94	0.91
Compress	0.90	0.99	0.94	0.92

最後我們以三次交叉驗證，來檢測二種資料格式的頁面清理結果，由表 11 來看，兩者表現相當，其 gap 相差不到 0.01。兩類模式的 Precision、Recall 及 F-Measure 均超過九成比率的準度，評估顯示出該兩類模型皆擁有不錯的清理效果。實驗結果可達到 68%～93%的 Precision、 91%～99%的 Recall 以及 0.81～0.91 的 F-Measure。

5.3 原始網頁與頁面清理擷取效能實驗

在此實驗中，本研究採用卅一份網頁做為擷取測試：ACM(02)、ADC(03,04,05)、Agents(00)、CIKM(02)、APWEB(04,05)、ER(00)、DASFAA(03,04)、DILS(06)、ECML/PKDD(04)、EDBT(04)、ICDE(04,05)、ICML(05)、IEEE(06)、IFIP(06)、IJCAI(05)、P2PIM(06)、SIGIR(03)、VLDB(98,99)、WAIM(06)、KDD(00,01,03)、WebDB(02)、WIDM(03)、WISE(03)。我們希望能分析並找出擷取規則適用於多樣性網頁的困難程度。

區分網頁複雜度最主要的原因在於有些網頁的複雜度和區塊結構有關，有的則是純粹以設計樣式有關；為了實證網頁複雜度，可能造成 SoftMealy 擷取上的困難，本研究將實驗網頁整體複雜度區分成 1-5 級的程度，再加上網頁設計上常用的四種組合[table, layout, list, list + layout]來實驗。經實驗數據顯示，SoftMealy 擷取時，在某些條件下，確實會因網頁內文雜訊及結構的複雜度而使得擷取正確率下降。

從表 12 的擷取數據來看，我們發現在 31 份網頁中，經由 SoftMealy 擷取正確記錄為 3370 筆，在未經清理的原始網頁其錯誤擷取數為 493 筆，錯誤擷取率 14.63%，正確擷取率 85.56%；經由頁面清理的錯誤擷取數為 145 筆，錯誤擷取率 4.3%，正確擷取率 95.81%。

表 12 SoftMealy 擷取實驗數據

範例	複雜度	手動標記次數	正確記錄	頁面未清理		頁面已清理		擷取錯誤原因
				錯誤記錄	正確率	錯誤記錄	正確率	
1	1	6	190	0	100 %	0	100 %	
2		4	94	0	100 %	0	100 %	
3		3	34	0	100 %	0	100 %	
4		3	40	0	100 %	0	100 %	
5		2	60	0	100 %	0	100 %	
6		3	28	0	100 %	0	100 %	
7		2	102	0	100 %	0	100 %	
8		2	42	0	100 %	0	100 %	
9		5	70	0	100 %	0	100 %	
10		2	480	0	100 %	6	98.75 %	SVM 清理錯誤
11		4	22	0	100 %	3	86.36 %	SVM 清理錯誤
12		4	34	0	100 %	5	85.29 %	SVM 清理錯誤
13		2	156	0	100 %	64	58.97 %	表格切割過於零碎
14		2	48	4	91.67 %	0	100 %	
15	2	2	268	0	100 %	6	97.76 %	LSD 切割錯誤
16		6	84	3	96.43 %	0	100 %	
17		3	44	7	84.09 %	0	100 %	
18	3	3	24	0	100 %	0	100 %	
19		2	204	32	84.31 %	0	100 %	
20		3	168	60	64.29 %	0	100 %	
21		3	96	43	55.21 %	0	100 %	
22		3	46	28	39.13 %	0	100 %	
23	4	4	90	0	100 %	0	100 %	
24		7	196	2	98.98 %	31	84.18 %	LSD 切割錯誤
25		3	40	10	75 %	0	100 %	
26	5	3	114	0	100 %	0	100 %	
27		4	50	0	100 %	16	68 %	SVM 清理錯誤
28		7	198	82	58.59 %	2	98.99 %	SVM 清理錯誤
29		6	124	64	48.39 %	2	98.39 %	SVM 清理錯誤
30		6	150	102	32 %	10	93.33 %	SVM 清理錯誤
31		3	74	56	24.32 %	0	100 %	
合計			3370	493	85.56 %	145	95.81 %	

當頁面清理階段正確，手動標示範例次數的減少，就可達到相當程度的正確率，因此在網頁複雜度低的擷取，未經清理的原始網頁與經由清理的效能，兩者間只有少量的差距，原則上並無不同。當網頁複雜度提高，因內文雜訊的影響，且使用者不感興趣的資料和欲擷取資料間的標籤相同而無法明顯區隔時，擷取的正確率就會明顯下降。由實驗數據來看，擷取正確率從未經頁面清理的 24.32% ~ 96.43% 到經由頁面清理的 100%，可看出清理過的擷取效果明顯優於未清理的結果。但是當頁面清理失誤時，如網頁複雜度低或是網頁複雜度高但內文雜訊和使用者感興趣的資料標籤不相同時，未清理擷取的正確率為 98.98%~100%，而清理的擷取正確率為 58.97%~98.75%，此時未清理的擷取效果反而優於清理過的結果。

倘若網頁複雜度高且內文雜訊和感趣興間的資料標籤是一致時，未清理擷取的正確率為 32%~58.59%，而清理的擷取正確率為 93.33% ~ 98.99%，此時清理的擷取效果又明顯優於未清理的結果。頁面清理主要發生的錯誤在於 (1) SVM 清理錯誤 (2) 表格切割過於零碎 (3) LSD 切割錯誤，此三項因素，將造成清理後的擷取成效低於未清理的主要原因，我們將在後續做討論。

最後一個階段實驗，採用已清理過的頁面，當作 SoftMealy 的訓練網頁，當頁面清理完全正確，網頁複雜度升高及雜訊內文彼此間標籤相同，SoftMealy 正確率將依網頁複雜度及雜訊內文標籤相同而逐漸下降；經由頁面清理後的 SoftMealy 對於複雜度網頁的擷取，其正確度起伏不大，平均可達到 95.81%以上。

5.4 問題討論

大部份的網站其資料呈現具有相當的結構或是有明顯界限符號，可以正確的擷取，但仍有部份網站屬性和屬性間沒有明顯區隔，此時仍需經背景知識的解析（自然語言處理技術來加強解譯），引入 NER 機制，辨識出現實環境中的個體物件，而無需人力標示，對於網站變異性大的網頁，應有助於提昇整體正確性。

頁面清理技術，係採用逐節點找尋，當資料呈現太多
, <p>, <tr>, <td>等標籤時，切割過於零碎，會直接影響到雜訊過濾的正確度；倘若 Page Cleaning 過濾錯誤，將會導致經由頁面清理的 SoftMealy 擷取效果遠低於單純的 SoftMealy 擷取。仍須加入自然語言語法來約束特徵的有效範圍。

6. 結論

在以往的資訊擷取研究，主要是以搜尋引擎或入口網站為主的資料整合，其擁有相同的 CGI 傳參方式使其具有共同的網頁結構或是相同網站下擁有類似的樣版模式為主的資料做整合，但跨網站的網頁內容擷取相對上就鮮少做探討。

本研究希望從網站網頁分類的角度來著眼，先識別出使用者需要的內容，經由頁面清理技術，將網頁複雜度降低，再做資訊內文的擷取。頁面清理對於樣式多變的 HTML 文件有淨化作用的擷取效果。本研究在實驗下的數據呈現出，在會議網站的接受論文網頁辨識下其回覆率可達到 80%，可減少使用者挑選網頁的負擔。另一實驗則是頁面清理擷取部份則是，未經清理的 SoftMealy 擷取，在經由頁面清理後其平均擷取錯誤率從原來的 14.63%降至 4.3%，而平均擷取正確率則從原來的 85.56% 提

昇至 95.81%，數據呈現出有相當程度改善效果，也證明網頁分類及頁面清理想法的可行性。

參考文獻

[1] C.-J. Lin's(LIBSVM), http://www.csie.ntu.edu.tw/%7Ecjlin/

[2] C.-N. Hsu and M.-T. Dung. Generating finite-state transducers for semi-structured data. Journal of Information Systems, Special Issue on Semi-structured Data, Volume 23, pp. 521-537, Aug 1998.

[3] C.-N. Hsu and C.-C. Chang. Finite-state transducers for semi-structured text mining. In Proceedings of IJCAI-99 Workshop on Text Mining: Foundations, Techniques and Applications, pp. 38-49, Stockholm, Sweden, 1999.

[4] D. Embley, Y. Jiang, and Y.-K. Ng. Record-boundary discovery in web documents. In Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD'99), pages 467–478, Philadelphia, PA, 1999.

[5] N. Cristianini and J. Shawe-Taylor. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press 2000.

[6] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods.

[7] SVM - Support Vector Machines, http://www.dtreg.com/svm.htm

[8] H. Cunningham, et al. Developing Language Processing Components with GATE Version, http://gate.ac.uk/sale/tao/index.html#annie