會議公告網站資訊擷取之研究

胡姝涵
國立中央大學 資訊工程所
92532007@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

資訊擷取（Information Extraction）的技術主要是將非結構化的資料，透過整理、篩選，加以整合成為結構化的資料。本篇論文主要針對國際性會議（International Conference）公告網站，擷取公告的國際會議資訊，包括會議名稱、會議地點、會議日期和論文接受日期。由於國際會議內容以純文字為主，加上會議內容的撰寫來自不同的公佈者，內容撰寫變化相當多，所以在資訊的整合與擷取上有一定的困難度，如何有效的擷取出正確的資訊，是一項有趣的挑戰。本篇論文運用機器學習的方式，讓電腦具有學習的能力，自動擷取來自不同公佈者公告的國際會議資訊，並且有不錯的效果。

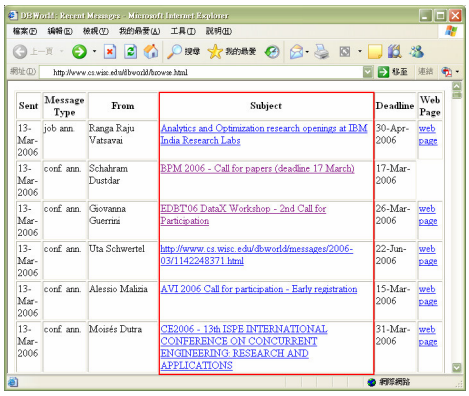
關鍵詞：資訊擷取、機器學習

1. 緒論

資訊擷取（Information Extraction）的技術主要是將非結構化的資料，透過整理、篩選，加以整合成為結構化的資料。資訊擷取的設計，最直接的方法是針對各個網站利用人工撰寫資訊擷取的方式，架構出符合此網站的資訊擷取系統，但由於網站的格式隨時有可能發生變更，或是因應不同作者架構出的網站格式不同，我們都必須修改撰寫不同的資訊擷取程式，這是非常不經濟的。因此，如何利用自動化的方式因應不同的網站格式來擷取網頁資訊，是設計資訊擷取程式最大的目標。自動化的資訊擷取設計，就要仰賴機器學習（Machine Learning）的方式，如何讓電腦具有學習的能力，從以往的經驗學習到知識和擷取規則，使得電腦本身具有擷取正確資訊的能力。

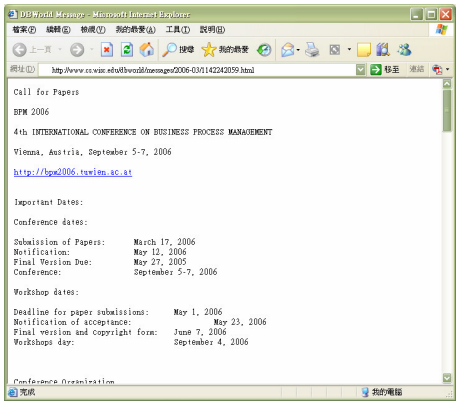
純文字的資訊擷取，主要是以自然語言的方式來處理，考慮句法分析和語意標籤，標記每個字的詞性，觀察字與字之間的相關性和句型文法結構等。過去撰寫這類型的資訊擷取程式，是採用直接由人工來撰寫擷取規則或是建立字典的方式，資訊擷取程式的撰寫者必須對要擷取的領域有一定程度的了解，透過人工的方式去觀察出適當的擷取規則或是利用字典的方式，把所有可能的擷取項目列於字典之中，再一一比對來找出所要擷取的項目，這二種方式都是非常直覺的，但卻缺乏彈性，當程式遇到沒有出現過的項目時，就無法正確的擷取出資訊，所以近年來研究朝向以機器學習的方式，讓電腦從以往的經驗學習到知識和擷取規則。雖然機器學習對擷取的正確率而言，沒有人工直接撰寫程式來得精確，但卻擁有無比的彈性，就開發成本和適用性而言，機器學習的方式是最佳的考量，且如何因應不同作者所架構網站格式的不同來考量資訊擷取能力，是設計資訊擷取程式最大的目標。

本篇論文所要擷取的資訊為國際性會議公告網站（DB World）上所發佈的會議名稱、會議地點、會議日期和論文接受日期。國際會議公告網站的網址為：http://www.cs.wisc.edu/dbworld/browse.html，如圖 1，其中的 Subject 為不同的佈告者以簡潔的短語所寫下的各個會議主旨，此為我們用來擷取國際會議名稱的來源。



Sent	Message Type	From	Subject	Deadline	Web Page
13-Mar-2006	job ann.	Ranga Raju Vatsavai	Analytics and Optimization research openings at IBM India Research Labs	30-Apr-2006	web page
13-Mar-2006	conf. ann.	Schahram Dustdar	BPM 2006 - Call for papers (deadline 17 March)	17-Mar-2006	
13-Mar-2006	conf. ann.	Giovanna Guerrini	EDBT'06 DataX Workshop - 2nd Call for Participation	26-Mar-2006	web page
13-Mar-2006	conf. ann.	Uta Schwertel	http://www.cs.wisc.edu/dbworld/messages/2006-03/1142248371.html	22-Jun-2006	web page
13-Mar-2006	conf. ann.	Alessio Malizia	AVI 2006 Call for participation - Early registration	15-Mar-2006	web page
13-Mar-2006	conf. ann.	Mozés Dutra	CE2006 - 13th ISPE INTERNATIONAL CONFERENCE ON CONCURRENT ENGINEERING: RESEARCH AND APPLICATIONS	31-Mar-2006	web page

圖 1 DB World 國際性會議的佈告網頁



Call for Papers

BPM 2006

4th INTERNATIONAL CONFERENCE ON BUSINESS PROCESS MANAGEMENT

Vienna, Austria, September 5-7, 2006

http://bpm2006.tuwien.ac.at

Important Dates:

Conference dates:

Submission of Papers: March 17, 2006
Notification: May 12, 2006
Final Version Due: May 27, 2005
Conference: September 5-7, 2006

Workshop dates:

Deadline for paper submissions: May 1, 2006
Notification of acceptance: May 23, 2006
Final version and copyright form: June 7, 2006
Workshops day: September 4, 2006

圖 2 會議論文徵求稿

由 DB World 網頁中，點入 Subject 的連結所呈現的內容為國際會議論文徵求稿，這包括網站連結、會議簡介、會議地點、會議日期、論文接受日期、等等其他的重要日期和一些相關的重要訊息，

如圖 2，會議論文徵求稿以純文字為主且內容結構會因不同的佈告者而有不同的呈現方式，此為我們擷取會議地點、會議日期和論文接受日期的主要擷取來源。

純文字資訊擷取的困難在於文字資料內容沒有明顯的標籤結構，加上不同的作者所架構的網站格式不同，寫法的風格也不同，語言的表達也沒有制式的標準，這些都是資訊擷取所要突破的地方。過去要撰寫此類的擷取規則，程式設計者除了要了解有什麼可用的特性可以做為擷取樣式或地標，還拼湊出準確率高的擷取規則，整個撰寫流程相當繁雜，而透機器學習的方式，我們只要著重在特徵值選取，對於文字資料的處理、規則的歸納都可由系統完成，更可以快速移植到不同的資訊擷取問題。

2. 相關研究

資訊擷取的規則可分為兩種，一種為單一插槽擷取（Single-slot），即同一網頁中一次只擷取一個項目，例如：若要擷取教授個人網頁中的教授名字和 E-mail，則必須分別產生兩個擷取規則；另一種為多插槽擷取（Multi-slot），即一個擷取規則同時擷取多個項目。本篇論文所要擷取的對象雖有四個，但均採單一插構的擷取規則，分別處理。

本篇論文使用的三種機器學習演算法，Naïve Bayes Classifier（機率）、SVM（統計）和 FOIL（推論歸納），都是廣泛使用的分類器。由於會議地點、日期的擷取都屬於 Named Entity Extraction，所以我們採用了 GATE ANNIE 做為我們的前置處理器。

ANNIE （A Nearly New Information Extraction System）由英國 Sheffield 大學所發表，是 plug-in 在 GATE （General Architecture for Text Engineering）[15][16][17]系統中的一個相當成熟的 NER（Name Entity Recognition）技術。GATE ANNIE 主要是運用 finite state algorithms 和 JAPE （Java Annotation Patterns Engine）language，來標記出文件中許多不同的 Named Entity，包括人名、地名、組織名稱、日期、時間等規則，它可以支援不同格式的文件，包括 Plain Text、XML、HTML、SGML、RTF、Email。JAPE Grammar 分成兩部份，left-hand-side(LHS)和 right-hand-side(RHS)，LHS 運用 regular expression operators('|', '*', '?', '+')組合所有可能要擷取的 pattern，RHS 可包含 Java code 或 macros，主要是針對 LHS 的 pattern 組合做進一步的處理。

3. 會議公告資訊擷取

本篇論文所要處理的擷取工作可細分為三種類型，第一種會議名稱的擷取，其輸入內容是 DB World 中的公告主旨，類似電子郵件的主旨型式，內容多以簡潔的短語來表明所要公告的事項，語法不拘；第二種會議地點和會議日期的擷取，其輸入內容為會議論文徵求稿，屬於半結構的純文字網頁，具有部份的規則性，第三種論文接受日期的擷取，其輸入內容與第二種是相同的資料來源，不同的地方是論文接受日期具有條例式的特性。

在正式介紹三個擷取工作之前，我們對於輸入文件均有前置處理的步驟：首先我們使用 GATE's ANNIE 的 NER 技術來找出文件中所有可能的地點和日期做為前處理的部份。由於 GATE's ANNIE 在標記地點和日期時會依標點符號做為所要處理文件的切割方式，因此同一地點可能會被切割為兩個地點。如以下範例所示。所以我們修改了 GATE's ANNIE 中地點和日期的 JAPE 規則擷取語法，讓相鄰的標示的結果能合乎我們的預期。

範例：

地名："San Diego, USA"

ANNIE 自動標記：**<Location>**San Diego**</Location>**, **<Location>**USA**</Location>**

修正後的結果：**<Location>**San Diego, USA**</Location>**

3.1 會議名稱

會議公告主旨通常非完整句而且也沒有標籤資訊，我們所能夠知道的是會議名稱是在會議公告主旨標題中的一小段連續性不被切割的字串，對此我們訓練資料格式的產生是採用 Sliding Windows 的方式，假設 window 最長為 w，則一個長度為 l 的主旨可以產生 $\sum_{i=1}^{w} i + (l-w)*w$ 片段。Tokenization 是以空白和標點符號做為資料的切割點，但連字號'-'和'@'不做切割點。舉例來說，給定範例資料 [CFP CoopIS 2006]，首先我們固定第一個 Token（CFP）可以產生三筆訓練資料：（CFP），（CFP CoopIS），及（CFP CoopIS 2006），接著固定第二個 Token（CoopIS ）分別產生（CoopIS）和（CoopIS 2006），最後再固定最後一個 Token（2006）並產生訓練資料（2006）。以上的每一筆訓練資料的形成，都是一個可能成為會議名稱的 Fragment。假設 CoopIS 2006 為我們所要擷取的會議名稱，則其標示為正例，其餘標示均為反例。

每一筆訓練資料我們取 Fragment 的前一個 Token、Fragment 的第一個 Token、Fragment 的第二個 Token、Fragment 的最後一個 Token 和 Fragment 的下一個 Token 做為判別會議名稱的屬性。如表 1 所示，每個 Token 均有 15 個特徵，分別測試一個 token 是否為日期、地點、文字、數值、混合文字數值、標點符號等。例如文字係考慮是否由英文字母所組成，我們又再細分為第一個字母大寫(W_Captial)和全部小寫字(W_Lower)兩種；數值部份為 0~9 的組成(W_Digit)；混合文字數值表一個 Token 中同時包含文字和數值(W_Mix)；我們還加入了三個特殊文字分別為 on、of 和 at；並且我們將 call for paper、Workshop 和 Proposals 等字詞以 W_CFP 和 W_Terminology 表示。而標點符號的部份，連字號'-'為文字中一部份，標點符號'@'在會議公告主旨中有英文單字'at'的意思，是辨別會議名稱的關鍵字，所以我們給予其單獨的特徵值。

NO	特徵值	特徵值註釋
01	W_CFP	call for paper、cfp
02	W_Date	日期
03	W_Location	地點
04	W_Mix	混合文字數值
05	W_Capital	第一個字母大寫
06	W_Digit	全部數字
07	W_Lower	全部小寫字
08	W_at	at
09	W_on	on
10	W_of	of
11	W_Terminology	Workshop、Proposals
12	P_Quote	括號中的資料
13	P_-	連字號‘-‘
14	P_Punct	標點符號
15	NULL	

表 1 會議名稱特徵值

3.2 會議地點、會議日期和論文接受日期

會議地點、日期和論文接受日期的資料來源是會議公告主旨連結中的會議邀稿文件，屬於半結構性的純文字網頁，但由於地點與日期係屬 Named Entity 擷取，透過 GATE's ANNIE 已可將文件中的地點及日期標示出來。不過由於徵求文件中可能出現多個地點，或是提及多個日期，因此我們的工作僅需加以分類，何者為會議地點，何者為會議日期，何者為論文接受日期。依擷取目標所在文件位置的不同，我們將會議地點、會議日期和論文接受日期分為兩類，會議地點和會議日期位置相似屬於同一類，而論文接受日期通常與論文截稿日與最後定稿日，有系統的呈現，因此我們又分屬於另一類。圖 3 為會議地點和會議日期的結構範例，圖 4 為論文接受日期結構範例。

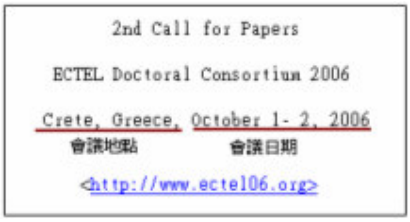


圖 3 會議日期、會議地點結構

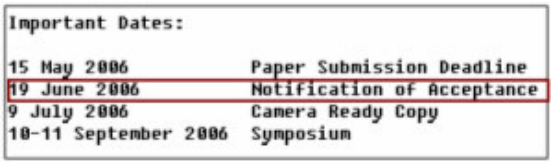


圖 4 論文接受日期結構

一般會議公告網頁中除了會議地點、會議日期和論文接受日期以外，還會包括 Program Committee 作者的國家所在地點或是主席的國家所在地點等，而日期方面也會存在許多的重要日期，所以整篇網頁中會夾雜著許多不相關的地點和日期，因此我們的工作即在眾多的地點和日期中以分類模式辨識出我們要的會議地點、會議日期和論文接受日期。

以會議地點為例子，對於 GATE ANNIE 所標示的每個地點，我們會分別取其前後 m 個 Token 做成其所代表的 fragment，並且取如表 2 所列之特徵，包括分為文字、數值、標點符號、日期、地點、空白、空行和歸位等。標點符號的部份，連字號’-‘、逗號’,’、冒號 ‘:’ 和句號 ‘.’在網頁中都具有重要的特徵，所以我們給予其單獨的特徵值，其餘的標點符號為另一個特徵值。

	特徵值	特徵值註釋
01	W_Capital	第一個字母大寫
02	W_Upper	全部大寫字
03	W_Lower	全部小寫字
04	W_Digit	全部數字
05	W_Num	數字中含有連字號
06	P_Quote	括號中的資料
07	P_Html	HTML 標籤
08	P_Comma	逗號’,’
09	P_Colon	冒號‘:’
10	P_Period	句號‘.’
11	P_Desh	連字號‘-‘
12	P_Other	其餘標點符號
13	C_Space	空白
14	C_CR	‘\r’
15	C_CN	‘\n’

表 2 會議地點和會議日期特徵值

論文接受日期具有條例式的結構，所以我們只憑會議地點和會議日期的 Token 特徵值，運用 Contextual Rule 來找出論文接受日期是不夠的，所以基於這樣的考量，我們另外增加兩個關鍵字（Notification、Acceptance），加上原有的會議地點和會議日期的 Token 特徵值做為論文接受日期的特徵值。如表 3。

	特徵值	特徵值註釋
15	W_Notification	Notification
16	W_Acceptance	Acceptance

表 3 論文接受日期特徵值

會議地點、會議日期和論文接受日期的訓練資料的格式是相同的，差別只在於特徵值設計的不同，我們在訓練和擷取時，是採用單一插槽（Single Slot）擷取的方式，所以雖然資料來源是相同的網頁，但我們擷取時會分別以會議地點、會議日期和論文接受日期進行三次的擷取。

4. 實驗與討論

本篇論文訓練和測試的資料來源是國際性會議佈告網站（DB World），共 150 筆會議公告主旨

做為會議名稱擷取的資料來源，其所對應的會議徵求論文稿共 150 個文件做為會議地點、會議日期、論文接受日期的資料來源，來做訓練和測試並比較實驗結果。分類演算法分別以統計為主的 Naïve Bayes Classifier、LibSVM（台灣大學林智仁教授）和以推論歸納為主的 FOIL 三種訓練模式。

我們實驗的方式採用三次的交叉驗證：隨機將資料分成三等份，取二等份為訓練資料，剩餘一等份做為測試資料，重複三次取其平均值。而我們評估是以準確率（precision）、涵蓋率（recall）和 F-Measure 來做為我們的評估方式。Precision 和 Recall 的計算方式，我們以混亂矩陣（Confusion Matrix）來計算，而 F-Measure 是準確率 Precision 和涵蓋率 Recall 二數值的協調平均值，三者的計算公式如下所示。

	計算接受	計算拒絕
實際接受	a	b
實際拒絕	c	d

表 4 混亂矩陣 (Confusion Matrix)

準確率： $\text{Precision}=\frac{a}{a+c}$

涵蓋率： $\text{Recall}=\frac{a}{a+b}$

F-Measure： $\text{F-Measure}=\frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$

演算法的參數設定的不同，對實驗結果的數據會有影響，Naïve Bayes Classifier 運用貝氏定理的理論，所以我們無法調整參數，SVM 方面我們使用 LibSVM 分別求出四組數據，第一組數據我們使用交差比對（cross validation）來找出最佳的訓練參數並搭配預設核心函數 Radial Basis Function（RBF）來求得實驗結果，第二組數據我們不使用參數，直接使用預設核心函數 RBF，第三組數據我們使用線性核心函數 Linear，第四組數據我們使用多項式核心函數 Polynomial，最後我們會從四組數據中選出一組最佳實驗結果。FOIL 方面我們固定 Gain 值的 threshold 為 0.7 和前提（antecedent）的容量為 5。

訓練和測試資料的格式，我們以兩種格式來實驗（sparse 和 compress），第一種 sparse 格式為，例如：在會議名稱中有 5 個屬性位置和 15 個特徵值，所以一筆 Fragment 會有 5*15=75 個屬性值，而會議地點和會議日期我們取前後各 5 個 token 和 15 個特徵值，所以一筆 Fragment 我們會有 10*15=150 個屬性值，最後論文接受日期的部分我們取前後各 4 個 Token 和 17 個特徵值，所以一筆 Fragment 會有 8*17=136 個屬性值。第二種 compress 格式是我們將每一個特徵值給予特定的優先等級，所以在會議名稱中有 5 個屬性，每一個屬性只會有一個特徵值，所以一筆 Fragment 會有 5 個屬性值，而會議地點和會議日期我們取前後 5 個 token，所以一筆 Fragment 我們會有 10 個屬性值，最後論文接受日期取前後 4 個 token，所以一筆 Fragment 會有 8 個屬性值。我們分別將四組的實驗結果說明如下。

4.1 會議名稱實驗結果

會議名稱來源資料是會議公告主旨，其資料的呈現是最沒有結構性的，完全依照每個佈告者的習慣來編寫，不同佈告者編寫的主旨內容有長有短，為了正確的擷取會議名稱，我們使用 Sliding Windows 的切割方式，所以一個會議公告主旨只會有一個正例，但會產生多個反例，若佈告者編寫的會議公告主旨太長則產生的反例會更多，這些都會影響實驗的結果。

如圖 5 和圖 6，我們以準確率 Precision 和涵蓋率 Recall 二數值的協調平均值 F-Measure 來看會議名稱的實驗結果，以整體來看，Compress 格式的結果較 Sparse 格式好，SVM Compress 的 F-Measure 值可達 89.91%，FOIL 為 77.11%。值得一提的是 FOIL 不同於統計型的 Naïve 和 SVM，FOIL 使用 Covering 的方式，由實驗結果發現 Precision 值很高，但 Recall 值卻不理想，使得間接影響到 F-Measure 值，這是因為會議公告主旨使用 Sliding Windows 切割後，所造成反例太多的原因。

FOIL 演算法會分別產生正例和反例數個 Rule，每個 Rule，不管是正例或反例都會計算出 Laplace accuracy，其計算方式為 Laplace accuracy = (classCounter+1) / (totalCounter+ 類別數)，[classCounter]為此 Rule 符合所屬類別（正例或反例）所能擷取此類別的個數，[totalCounter]為不管正例或反例，此 Rule 所能擷取的個數，[類別數]為分為幾類，以我們的例子來說類別數為 2，即正例和反例 2 個類別。當反例很多時，則反例 Rule 的 Laplace accuracy 就會很高，則當此 Rule 同時能擷取正例和反例的類別時，會選擇擷取 Laplace accuracy 值高的，所以在會議名稱中，因為反例太多，反例 Rule 的 Laplace accuracy 值就會比正例 Rule 的 Laplace accuracy 值高，加上會議名稱為不具結構性，所以其產生的反例 Rule 也不具結構性，這些都會造成 Recall 值受到影響。

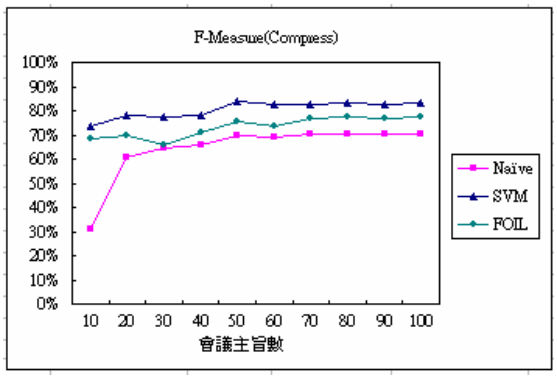


圖 5 會議名稱 Naïve 、SVM 和 FOIL 的 F-Measure (Compress)值比較

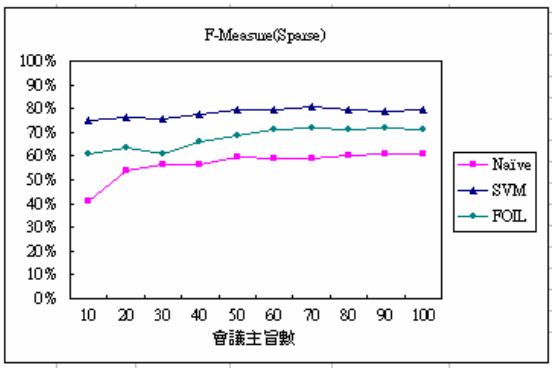


圖 6 會議名稱 Naïve 、SVM 和 FOIL 的 F-Measure (Sparse)值比較

4.2 會議地點實驗結果

會議地點的資料來源純文字檔的論文徵求稿，一般佈告者習慣在公佈會議地點的附近也一起公佈會議日期，所以我們可以依照這樣的特性擷取正確的會議地點，但免不了還是會存在一些佈告者以不同的格式來公告國際會議資訊，且一篇網頁中除了會議地點以外，還會存在主席和作者的國家地點，飯店的所在地等資訊，這些都是影響會議地點的擷取效果，不過由於仍具有部份的結構性，所以我們將會議地點歸類為半結構性的資料。

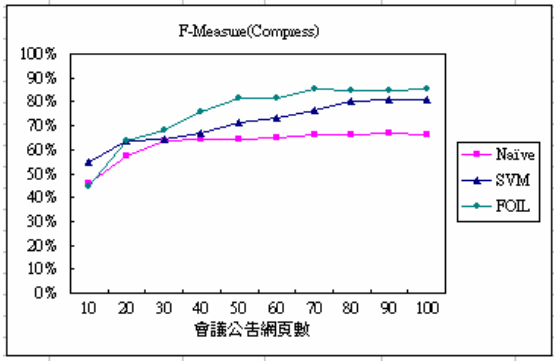


圖 7 會議地點 Naïve 、SVM 和 FOIL 的 F-Measure (Compress)值比較

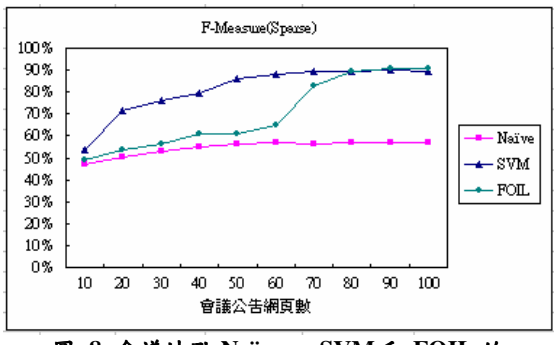


圖 8 會議地點 Naïve 、SVM 和 FOIL 的 F-Measure (Sparse)值比較

4.3 會議日期實驗結果

會議日期的資料來源與會議地點相同，是具有半結構性的資料，但是比較起來會議日期的結構性較會議地點差，因為會議日期的結構性同樣受到佈告者編寫習慣的不同，且整篇網頁中除了公佈的會議日期以後，還有具條例式的會議公告重要日期，其中還有不少是佈告者以一般自然語言型式來提醒大家相關的註冊日期、住宿申請日期等各項重要訊息的最後期限，所以相較於會議地點，會議日期是較不具結構性的資料。

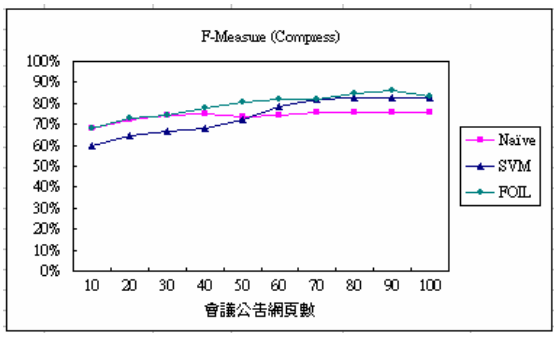


圖 9 會議日期 Naïve 、SVM 和 FOIL 的 F-Measure (Compress)值比較

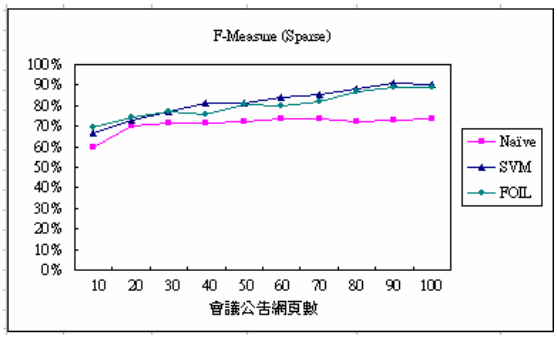


圖 10 會議日期 Naïve 、SVM 和 FOIL 的 F-Measure (Sparse)值比較

如圖 9 和圖 10 會議日期的實驗結果，由實驗結果顯示，Naïve 仍然是以 Compress 格式較好，其 F-Measure 值為 68% ~ 76%，同樣的 SVM 和 FOIL 皆以 Sparse 格式的實驗結果較好。FOIL(Sparse)最高可達 88.98%，SVM(Sparse)最高可達 90.65%。

4.4 論文接受日期實驗結果

同樣是日期結構，論文接受日期不同於會議日期的地方是，論文接受日期的位置通常在網頁作者公佈重要日期的地方，且其結構性呈條例式的方式呈現，所以相較於會議日期是較具結構性的。

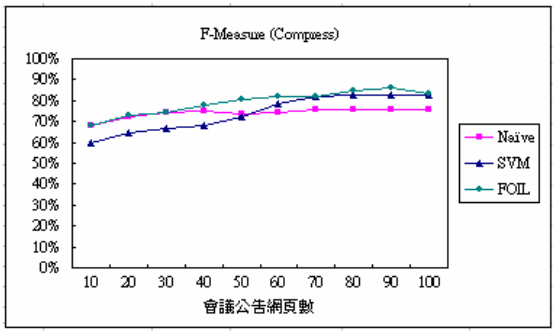


圖 11 會議日期 Naïve 、SVM 和 FOIL 的 F-Measure (Compress)值比較

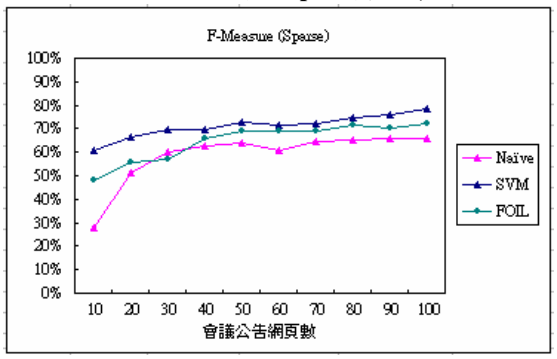


圖 12 會議日期 Naïve 、SVM 和 FOIL 的 F-Measure (Sparse)值比較

如圖 10 和圖 11 論文接受日期的實驗結果，以整體來看 Naïve、SVM 和 FOIL 皆是以 Compress 和格式的結果較 Sparse 格式好，Naïve (Compress)為 43%~75%，SVM (Compress)為 62%~88%，FOIL (Compress)為 61%~86%。

4.5 討論

表 5 為四組不同資料類型的 F-Measure 值選出實驗結果最高的 Compress 格式，而表 6 為 Sparse 格式中 F-Measure 值的實驗結果最高的。SVM 在經過調整參數值和核心函數轉換後和使用推論歸納的 FOIL 其實驗結果是最好的，而 Naïve 的擷取結果就顯得較差。

FOIL 在會議名稱的擷取時，由於反例太多且不具結構化的情況下，使得產生的反例 Rule 也不具結構性，而影響擷取結果。

	名稱	地點	日期	論文接受日期
Naïve	70.14%	66.67%	75.95%	75.79%
SVM	82.91%	81.07%	82.82%	88.14%
FOIL	77.11%	85.60%	82.86%	85.81%

表 5 綜合 F-Measure (Compress)實驗結果

	名稱	地點	日期	論文接受日
naïve	60.78%	57.05%	73.29%	66.60%
SVM	79.34%	90.39%	90.46%	78.65%
FOIL	70.93%	90.93%	88.98%	71.91%

表 6 綜合 F-Measure (Sparse)實驗結果

5. 結論與未來展望

現今網路上的資訊大多數仍以純文字的自然語言呈現，如：期刊、電子報、工作記錄等，就連我們每天都使用的電子郵件也是以自然語言的方式來表達，雖然人們容易閱覽且易於擷取出重要資訊，但對於讓電腦能夠理解，還有很大的進步空間。本篇論文運用機器學習的方式，加上領域背景知識的特徵值選取，擷取出四種來自不同網頁且不同結構的會議公告資訊，充分的解決了傳統對於非結構化純文字的擷取，必須仰賴人工自行撰寫擷取程式的困擾。

本篇論文在不具結構性的會議名稱中，由於沒有明顯會議名稱的起始和結束邊界，所以我們採用 Sliding Windows 的切割方式，在 Compress 和 Sparse 格式下，Naïve 、SVM 和 FOIL 的 F-Measure 值皆沒有很好的擷取結果，因為 Sliding Windows 的切割方式會產生太多的反例，再加上會議名稱的內容並無一特定格式，這些都會影響分類的準確率。若我們能夠利用更多的背景知識來減少大量的反例或將資料處理更具結構性，則擷取率便會提高，但也因此會使擷取程式更受限於某領域的擷取。

對於處理無明顯邊界的純文字切割，與 Sliding Windows 相對的另一個方法是使用 Hidden Markov Model（HMM）的方式，HMM 近幾年來被引入資訊擷取的研究，運用 HMM 的方式可以解決 Sliding Windows 切割方式所產生過多反例的弱點。所以會議名稱的部份在未來的方向可以運用 HMM 演算法，來因應會議公告主旨內容變化量大且產生過多反例的問題。

參考文獻

[1] D. Freitag. *Information Extraction from HTML: Application of a General Machine Learning Approach*. In Proceedings of the Fifteenth national Conference on Artificial Intelligence, pages 517–523, 1998

[2] M.E. Califf, and R.J. *Mooney. Relational learning of pattern-match rules for information extraction.* In Proceedings of the 16th National Conference on AI, 328-334, 1999.

[3] I. Muslea, S. Minton, and C. Knoblock, *A hierarchical approach to wrapper induction*. In Proceedings of 3rd International Conference on Autonomous Agents（Agents-99）,pp. 190-197, Seattle, Washington,1999

[4] C.-N. Hsu. *Initial Results on Wrapping Semi-structured Web Pages with Finite-State Transducers and Contextual Rules*. In Proceedings of AAAI-98 Workshop on AI and Information Integration, Technical Report WS-98-01. 1998.

[5] C.-H. Chang and S.-C. Lui. *IEPAD: Information Extraction Based on Pattern Discovery.* In Proceedings of 10th International Conference

on World Wide Web, pp. 681-688, 2001

[6] J. Wang, and F.H. Lochovsky. *Data Extraction and Label Assignment for Web Databases*. In Proceedings of the twelfth international conference on Wide Web, Page 187 - 96, 2003.

[7] B. Liu, R. Grossman, and Y. Zhai. *Mining Data Records in Web Pages*. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Date Mining (KDD'03), Page 24 - 27, 2003

[8] GATE – *An Application Developer's Guide* http://www.dcs.shef.ac.uk/~valyt Department of Computer Science University of Sheffield, UK. 19 July 2004

[9] Weka The University of Waikato http://www.cs.waikato.ac.nz/ml/weka/

[10] W. Coenen, F. *LUCS-KDD implementations of the FOIL, PTM and CPAR algorithms*, http://www.cxc.liv.ac.uk/~frans/KDD/Software/FOIL_PRM_CPAR/,Department of Science, The University of Liverpool, UK. (2004)

[11] T. M. Mitchell, Carnegie Mellon University, Machine Learning

[12] J. Han, Micheline Kamber, Data Ming concepts and Techniques

[13] LIBSVM, http://www.csie.ntu.edu.tw/~cjlin/libsvm/