行政院國家科學委員會專題研究計畫　期中進度報告

詞彙語意關係 之自動標注—以中英平行語料庫為基礎

(2/3)

計畫類別： 個別型計畫
計畫編號： NSC92-2411-H-002-061-
執行期間： 92 年 08 月 01 日至 93 年 07 月 31 日
執行單位： 國立臺灣大學外國語文學系暨研究所

計畫主持人： 高照明
共同主持人： 劉昭麟

報告類型： 精簡報告
報告附件： 出席國際會議研究心得報告及發表論文
處理方式： 本計畫可公開查詢

中　華　民　國 93 年 5 月 13 日

國科會專題研究計畫期中報告

計畫編號：NSC92-2411-H-002-061

計畫名稱：詞彙語意關係之自動標注

—以中英雙語平行語料為基礎（2/3）

執行單位：國立台灣大學外國語文學系

計畫主持人：高照明

協同主持人：政大資科系劉昭麟

計畫助理：黃子桓，林語君，吳昊極，戴士強，

劉彥廷，張席維，江佳恩，陸鳳玥，余世傑

計畫摘要

本計畫希望利用中英平行語料裡英文的句法結構與中英文詞彙知識庫發展一個能夠自動標示中文詞彙語意關係的系統。英文方面我們將利用現有的英文詞類標注程式（Brill1997）句法剖析器 （Link Parser, Cass Parser），英文詞彙知識庫（Wordnet），英文語法資料庫 (Comlex)。中文方面我們將利用中研院詞庫小組的分詞與詞類標記程式及中文語法資料庫，董振東先生的知網(Hownet)及訊息結構，同義詞詞林，教育部國語會國語辭典等資源。首先我們將中英文文件分句。接著將中文文件分詞與標記詞類。透過電子辭典與統計的訊息，我們可以得到部分詞彙對應及句子對應的關係。藉由這些訊息與英文的語法結構，我們可以推斷中文詞組的界限。一旦詞組的界限找到，我們將結合英文對應詞在 Wordnet 之間的詞彙語意關係，中文知網(Hownet)，訊息結構，與中研院詞庫小組的中文語法資料庫確定詞組內詞彙之間的關係究竟是下列哪一種關係，（一）謂語與論元關係（二）主語與謂語關係（三）上下位關係（四）部件與整體關係（五）同義關係（六）反義關係等。由於文章通常都圍繞在某一主題，相關的詞彙常會一再出現，透過詞在篇章裡的訊息及 Hownet 與同義詞詞林我們可以進一步確定出這些詞的關係。

本計畫到目前爲止執行順利得到預期的成果。包括下列幾項。

1. 設計一個依據雙語辭典非精確比對爲主的雙語詞彙對應程式。由於我們採用幾項賦予權值的新方法即使在只有部分比對到的情形下也能從中英平行語料得到不錯的正確率。我們目前還在繼續改良正確率。
2. 透過英文的詞組結構與雙語詞彙對應，我們也得到部分詞組結構的對應。但由於翻譯無法靠雙語詞彙對應找到全部的詞組對應，所以還必須靠其它的訊息來解決詞組結構的對應的問題。
3. 利用雙語辭典和 wordnet，中文分詞程式，我們設計了一個工具可以輸入中文後分詞，翻譯成最常用的英文，然後以 wordnet 內建的訊息找詞之間的語意關係。
4. 利用 Hownet 雙語義元的特性找出中文文章及英文文章裡面具有語意關係的詞。
5. 在中英詞對應的基礎上應用 Wu (1997)所提出的 Transduction Inversion Grammar,作爲未來發展雙語形式文法的初步基礎。

我們實做並比較幾種統計式的雙語辭彙擷取演算法，包括 Fung and Church (1994,1997) Kvec 以及 Brown (1993) IBM Model 1,但效果都不理想。除了統計式

的辭彙擷取，我們也嘗試從已知的字典中擷取更多的訊息。字典是人工所編，因此正確率很高，然而字典字數不足、應用太少，難以面對各式各樣的文類與文體，因此傳統的精確比對難以充份利用字典的優勢，我們改以嘗試部份比對的技巧，希望能提高字典的使用。

基於字典的部份比對權值計算演算法設計如下：

假設兩個語言爲 C[1..n] 和 E[1..m]，其中 C[1]、C[2]…C[n]、E[1]、E[2]…E[m] 皆爲該語言中的「字彙」。要進行兩個語言的轉換，我們需要 C[i] 和 E[j]，1 <= i <= n and 1 <= j <= m 兩兩的相關程度，我們用 R[i, j] 表示 C[i] 和 E[j] 的相關程度。

計算 R[i, j] 有很多方法，例如基於統計的 MI、T-score 等，而另一個方法就是基於字典的部份比對權值計算。經過字典的蒐尋，我們可以對字典所對應到的 C[i] 和 E[j] 有很大的信心，而給與 L[i, j] 一個很大的權值。然而由於現實的緣故，我們有理由相信不論字典規模多大，文章中仍然有機會出現字典所沒有的字彙，而出現的比例與兩篇文章的翻譯方式、文章的文體類別…等有關。因此我們仍必需在字典找不到對應時，有其它的方法決定 C[i] 和 E[j] 的相關程度，這個方法就是部份比對的權值計算。

現在我們假定 C 爲中文、E 爲英文。我們用 length(W[i]) 表示 W[i] 這個字彙的長度 (在這裡 W 可爲 C 或 E)，W[i][u] 表示 W[i] 這個字彙的第 u 個「字」，顯然 1 <= u <= length(W[i])。在這裡，對中文而言就是「單一的字」，對英文而言就是「字母」。則我們可以在字典中查詢 W[i] 中的部份字串，這個查詢的結果可以給我們部份的信心，使我們能給定一個相關的程度。例如，在字典中查不到 unbuffered，然而部份字串 buffer 查到的機會顯然較大，因此我們有部份的信心 unbuffered 和 buffer 所查到的對應中文有關。然而這個部份信心和部份字串的長度有關，例如，查到 ere 對應的信心顯然比查到 buffer 對應的信心還少，因此這個信心是和部份字串所占總體長度的比例有關的，部份字串越長，我們的信心就越大，反之亦然。而這個信心的估計方式也可以適用於查詢英文所對應到的中文字串，例如查詢英文 field 得到「場地」，然而在我們的文章中，只有「場」這個字彙，因此我們有部份的信心 field 和「場」是相關的。我們令 B[i, j] 爲 C[i] 某個部份字串和 E[j] 某個部份字串的信心指數的最大值，爲方便，我們令 0 <= B[i, j] <= 1，顯然當 B[i, j] 爲 1 時表示 E[j] 能從字典查找到 C[j]；而 B[i, j] 爲 0 時表示沒有任何部份字串顯示 C[i] 和 B[j] 有相關。

當我們從英文查找中文時，有可能在一個句子中找到兩個以上的中文字彙符合查找的結果，這時候我們用另一個方法給出英文對兩個中文的權值關係。首先，在多數機會的中英翻譯中，中文句法和英文句法是照順序對譯的，因此我們可以相信，若英文和中文在該句中的位置相近，則我們有較大的信心支持這兩個字彙相關。我們令 D[i, j] 表示 C[i] 與 E[j] 位置相近的信心程度，同樣的，0 <= D[i, j] <= 1，若相對的兩個字彙在完全相同的兩個相對位置上，則其權值爲 1，而若相對的兩字彙一在句首一在句尾則其權值爲 0。

我們令 V[i, j] = base_cvalue * C[i, j] + base_dvalue * D[i, j]，其中 base_cvalue 和 base_dvalue 是 C[i, j] 和 D[i, j] 的基本權值，因爲 C[i, j] 和 D[i, j] 必然小於等於 1，因此 0 <= V[i, j] <= base_cvalue + base_dvalue。而 base_cvalue 和 base_dvalue 的比例與大小則需經過資料蒐集分析，從統計的結果得知。
從實作的結果，我們發現部份比對的理論架構對於英文複合字所成的新字，例如 un-、im- 等反義字，或 e-mail 等縮寫複合字會有較大的效果，而若是較長的單字，則其單字的部份字串是另一單字的機率大爲提高，因此有必要對大量的資料做實測，找到較佳的控制參數值。

另外我們也提出如何透過語法結構，以部分比對和權值的計算得到較多的雙語詞彙對應。首先我們必需有兩份資訊，第一是英文parsed 的結果，第二是中、英文各字的對應分數。

我們使用EN[1..length EN] 和CH[1.. length[CH]] 來表示英文和中文， 其中EN[i] 表示英文的第i 個詞(單字)， CH[i] 表示中文的第i 個詞(由中文分句程式而得的詞)。

接著，我們使用V en_{start},en_{end},ch_{start},ch_{end} 來表示當EN[en_{start}..en_{end}] 和 CH[ch_{start}..ch_{end}] 對應時所得到的分數， 則我們的目標就是找出在EN[1.. length EN] 和CH[1.. length CH] 中能使V 最大的en_{start}、en_{end}、ch_{start}、ch_{end}。

我們首先從英文parsed 的結果來觀查： 我們使用mlparser 來處理英文， 結果爲一syntax tree， 其中每一個leaf 都對應到該英文句子中的一個詞(單字)。因此我們的第一步驟即是要找出這些leaf 所對應的中文。

對於每一個leaf EN [i] ， 我們先找出所有V i , i ,j j ,1≤ j≤length CH 不爲零的j ， 並將其依V 值重新排序，假設爲{ $j_1 j_2 ... , j_n$} ，並令 ,j_k所對應的V i , i , j_k, j_k 值爲v_k。因

n

此可知， $v_1 \geq v_2 \geq ... \geq v_n$。令dev = $\frac{1}{\text{Sigma}_{k=1}}$ $v_1 - v_k$ 。令C 爲EN [i] 所對應中文的集

n

合， 則若v_k dev≥v_1且j和min C 或max C 的距離不超過一定值， 則我們將j_k加入到C 裡。在此

k

「定值」考慮英文的複 合名詞， 中文有部份機會譯爲「□□ 的□□ 」， 因此給定爲2。當C 計算完畢， 則我們可以令EN [i] 所對應的中文爲CH [min C .. max C] ， 而令 V i ,i , min C , max C =Sigma$_{e \ C}$V i ,i ,e ,e 。

而中文 parser 的研究部份，我們實作了隨機倒置語法 Transduction Inversion Grammar 的演算法模型。隨機倒置語法的演算法如下：
假設中文、英文文章爲 C[1..n_C]，E[1..n_E]，其中 C[1]、C[2]、… E[1]、E[2] 皆爲中、英文詞彙。則我們可由 k-vector 或基於字典的部份比對模型得到 C[i] 對應 E[j]，1 <= i <= n_C，1 <= j <= n_E 的信心值，以 F(i, j) 表示。我們以 V^{[]}(s_C, e_C, s_E, e_E) 表示 C[s_C]C[s_C+1]...C[e_C] 順向對應 E[s_E]E[s_E+1]...E[e_E] 的權值，而 V^{\{\}}(s_C, e_C, s_E, e_E) 表示 C[s_C]C[s_C+1]...C[e_C] 反向對應 E[s_E]E[s_E+1]...E[e_E] 的權值。定義：
V(s_C, e_C, s_E, e_E) = max (V^[](s_C, e_C, s_E, e_E), V^{}(s_C, e_C, s_E, e_E))
V^[](s_C, e_C, s_E, e_E) = max_{s_C <= i <= e_C, s_E <= j <= e_E,
(e_C-i)(i-s_C)+(e_E-j)(j-s_E) != 0} V(s_C, i, s_E, j)V(i, e_C, j, e_E)
V^{}(s_C, e_C, s_E, e_E) = max_{s_C <= i <= e_C, s_E <= j <= e_E,
(e_C-i)(i-s_C)+(e_E-j)(j-s_E) != 0} V(s_C, i, j, e_E)V(i, e_C, s_E, j)
而由初值
V(i-1, i, j-1, j) = F(i, j) 1 <= i <= n_C, 1 <= j <= n_E
V(i-1, i, j, j) = F(i, NULL) 1 <= i <= n_C
V(i, i, j-1, j) = F(NULL, j) 1 <= i <= n_C
其中，F(i, NULL) 和 F(NULL, j) 表示該 C[i] 或 E[j] 沒有對應的詞彙，實作上可以給定一足夠小的非零值以代表。有了上述的遞迴式，我們可以用動態規劃的方式，求出整理中英文配對權值最高的結果，再依此結果搭配英文 parser 來剖悉中文的語法樹。
實作的結果中，這個模型對於「詞對詞」的翻譯方式有相當好的結果，然而由於中英文語言差異使然，真正能完全詞對詞翻訪的情形並不多見，較合理的翻譯方式是詞組對應詞組。而詞的界限也是該模型較難克服之處，因爲 F(i, NULL) 和 F(NULL, j) 給定一很小的值，因此很多無對應的功能詞會被前後的詞包含，而若 F(i, NULL) 和 F(NULL, j) 的值給定過大，又容易使結果大量出現中、英文對應皆爲空字的情形發生。

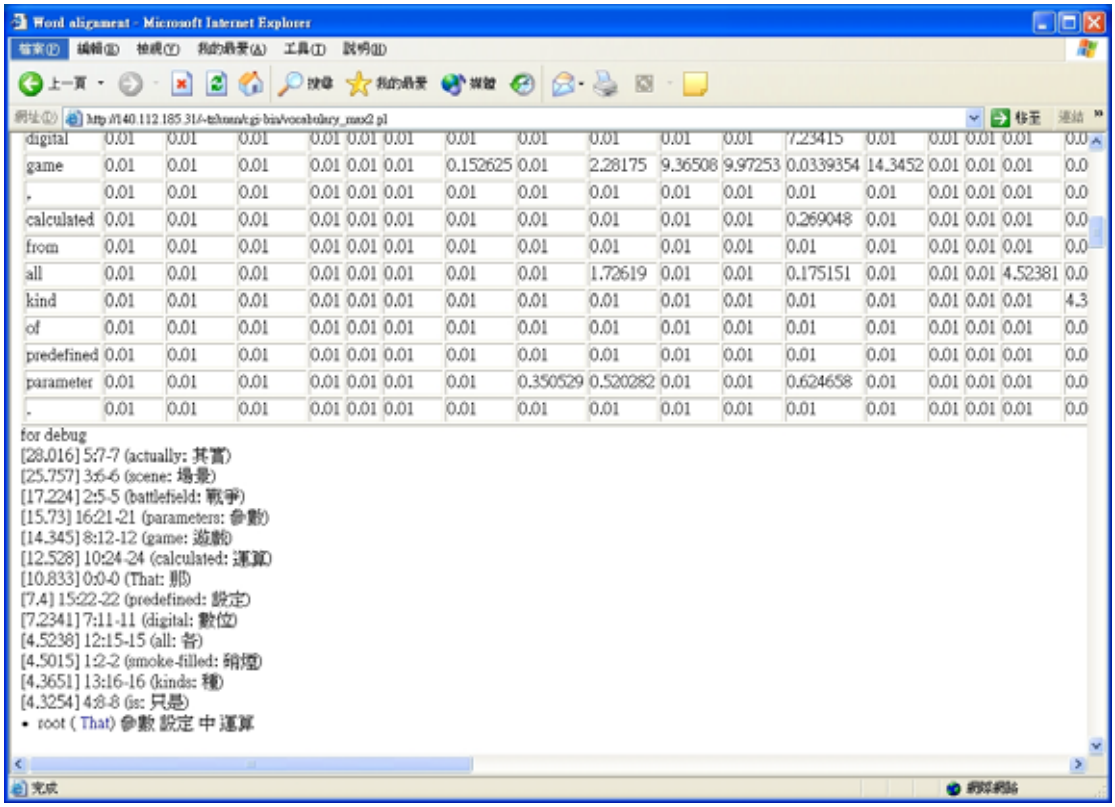
有了以上的基礎，我們接下來的目標在於改善上述論理的缺失，研究如何利用不同的理論模型來搭配使用，以得到更高的正確率和可用性。除此之外，我們也希望能另闢蹊徑，在統計的方法之外，找尋另一個雙語資料處理的方向。

附錄：

圖 1 利用部分比對與權值計算得到雙語詞彙對應的過程

	那	裡面	硝煙	遍布	的	戰爭	場景	其實	只是	一	場	數位	遊戲	,	在	各	種
that	10.8333	4.88095	0.01	0.01	0.01	0.01	0.01	3.57143	3.21429	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
smoke-filled	0.01	0.01	4.50149	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
battlefield	0.01	1.06421	0.436508	0.01	0.01	17.2243	0.840246	0.01	0.01	0.01	1.35967	0.01	0.01	0.01	0.01	0.01	0.0
scene	0.01	0.129552	0.01	0.01	0.01	0.01	25.757	2.14286	0.01	3.92857	4.76621	0.01	0.01	0.01	0.01	0.01	0.0
is	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	4.3254	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
actually	0.01	0.01	0.01	0.01	0.01	0.01	0.01	28.0159	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
a	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
digital	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	7.23415	0.01	0.01	0.01	0.01	0.0
game	0.01	0.01	0.01	0.01	0.01	0.01	0.152625	0.01	2.28175	9.36508	9.97253	0.0339354	14.3452	0.01	0.01	0.01	0.0
,	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
calculated	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.269048	0.01	0.01	0.01	0.01	0.0
from	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
all	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	1.72619	0.01	0.01	0.175151	0.01	0.01	0.01	4.52381	0.0
kind	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	4.3
of	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
predefined	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0

圖 2 利用部分比對與權值計算得到雙語詞彙對結果



digital	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	7.23415	0.01	0.01	0.01	0.01	0.0
game	0.01	0.01	0.01	0.01	0.01	0.01	0.152625	0.01	2.28175	9.36508	9.97253	0.0339354	14.3452	0.01	0.01	0.01	0.0
,	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
calculated	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.269048	0.01	0.01	0.01	0.01	0.0
from	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
all	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	1.72619	0.01	0.01	0.175151	0.01	0.01	0.01	4.52381	0.0
kind	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	4.3
of	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
predefined	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0
parameter	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.350529	0.520282	0.01	0.01	0.624658	0.01	0.01	0.01	0.01	0.0
.	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0

圖 3 利用英文剖析器與中英文詞彙對應關係得到的詞組對應

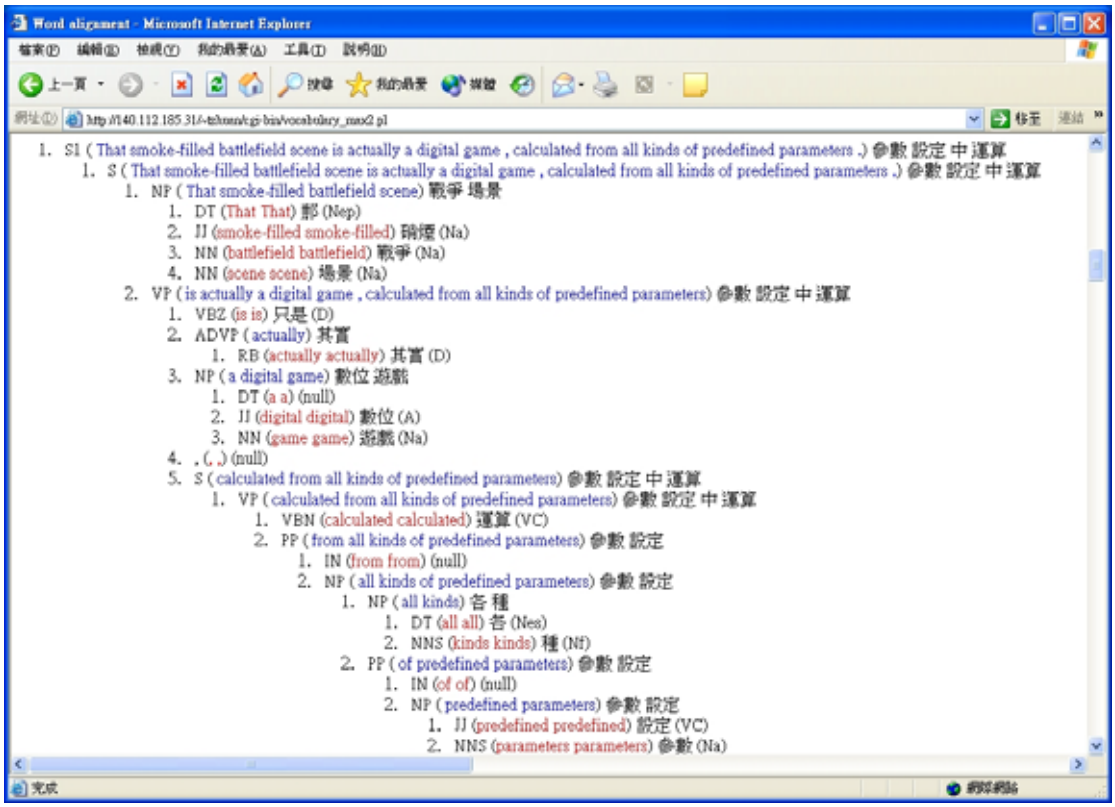
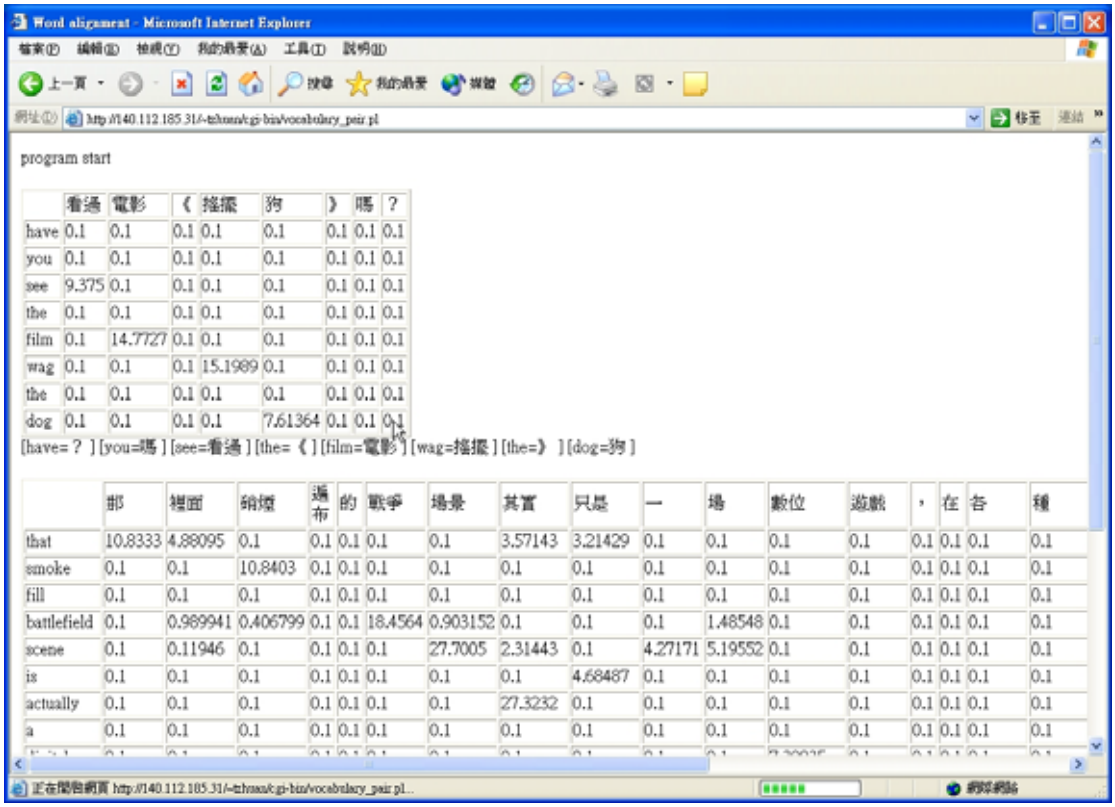


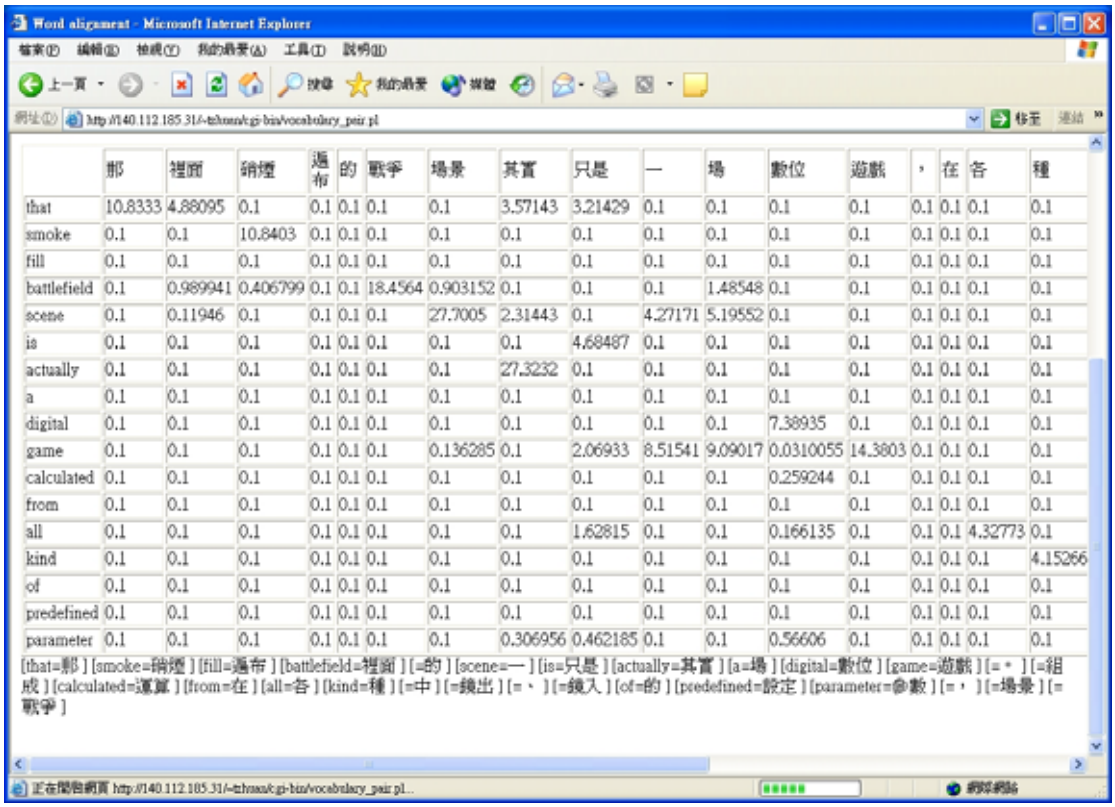
圖 4 利用雙語詞彙對應與動態規劃及 Transduction Inversion Grammar 得到的結果



	看過	電影	《	搖擺	狗	》	嗎	?
have	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
you	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
see	9.375	0.1	0.1	0.1	0.1	0.1	0.1	0.1
the	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
film	0.1	14.7727	0.1	0.1	0.1	0.1	0.1	0.1
wag	0.1	0.1	0.1	15.1989	0.1	0.1	0.1	0.1
the	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
dog	0.1	0.1	0.1	0.1	7.61364	0.1	0.1	0.1

	部	裡面	硝煙	遍布	的	戰爭	場景	其實	只是	一	場	數位	遊戲	，	在	各	種
that	10.8333	4.88095	0.1	0.1	0.1	0.1	0.1	3.57143	3.21429	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
smoke	0.1	0.1	10.8403	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
fill	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
battlefield	0.1	0.989941	0.406799	0.1	0.1	18.4564	0.903152	0.1	0.1	0.1	1.48548	0.1	0.1	0.1	0.1	0.1	0.1
scene	0.1	0.11946	0.1	0.1	0.1	0.1	27.7005	2.31443	0.1	4.27171	5.19552	0.1	0.1	0.1	0.1	0.1	0.1
is	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	4.68487	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
actually	0.1	0.1	0.1	0.1	0.1	0.1	0.1	27.3232	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
a	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1

圖 5 利用雙語詞彙對應與動態規劃及 Transduction Inversion Grammar 得到的結果



	郵	裡面	硝煙	遍布	的	戰爭	場景	其實	只是	一	場	數位	遊戲	，	在	各	種
that	10.8333	4.88095	0.1	0.1	0.1	0.1	0.1	3.57143	3.21429	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
smoke	0.1	0.1	10.8403	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
fill	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
battlefield	0.1	0.989941	0.406799	0.1	0.1	18.4564	0.903152	0.1	0.1	0.1	1.48548	0.1	0.1	0.1	0.1	0.1	0.1
scene	0.1	0.11946	0.1	0.1	0.1	0.1	27.7005	2.31443	0.1	4.27171	5.19552	0.1	0.1	0.1	0.1	0.1	0.1
is	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	4.68487	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
actually	0.1	0.1	0.1	0.1	0.1	0.1	0.1	27.3232	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
a	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
digital	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	7.38935	0.1	0.1	0.1	0.1	0.1
game	0.1	0.1	0.1	0.1	0.1	0.1	0.136285	0.1	2.06933	8.51541	9.09017	0.0310055	14.3803	0.1	0.1	0.1	0.1
calculated	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.259244	0.1	0.1	0.1	0.1	0.1
from	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
all	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	1.62815	0.1	0.1	0.166135	0.1	0.1	0.1	4.32773	0.1
kind	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	4.15266
of	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
predefined	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
parameter	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.306956	0.462185	0.1	0.1	0.56606	0.1	0.1	0.1	0.1	0.1

[that=郵] [smoke=硝煙] [fill=遍布] [battlefield=裡面] [=的] [scene=一] [is=只是] [actually=其實] [a=場] [digital=數位] [game=遊戲] [=，] [=組成] [calculated=運算] [from=在] [all=各] [kind=種] [=中] [=鏡出] [=、] [=鏡入] [of=的] [predefined=設定] [parameter=參數] [=，] [=場景] [=戰爭]

圖 6 輸入中文用 Hownet 比對義元

圖 7. 輸入中文用 Hownet 比對義元得到的結果

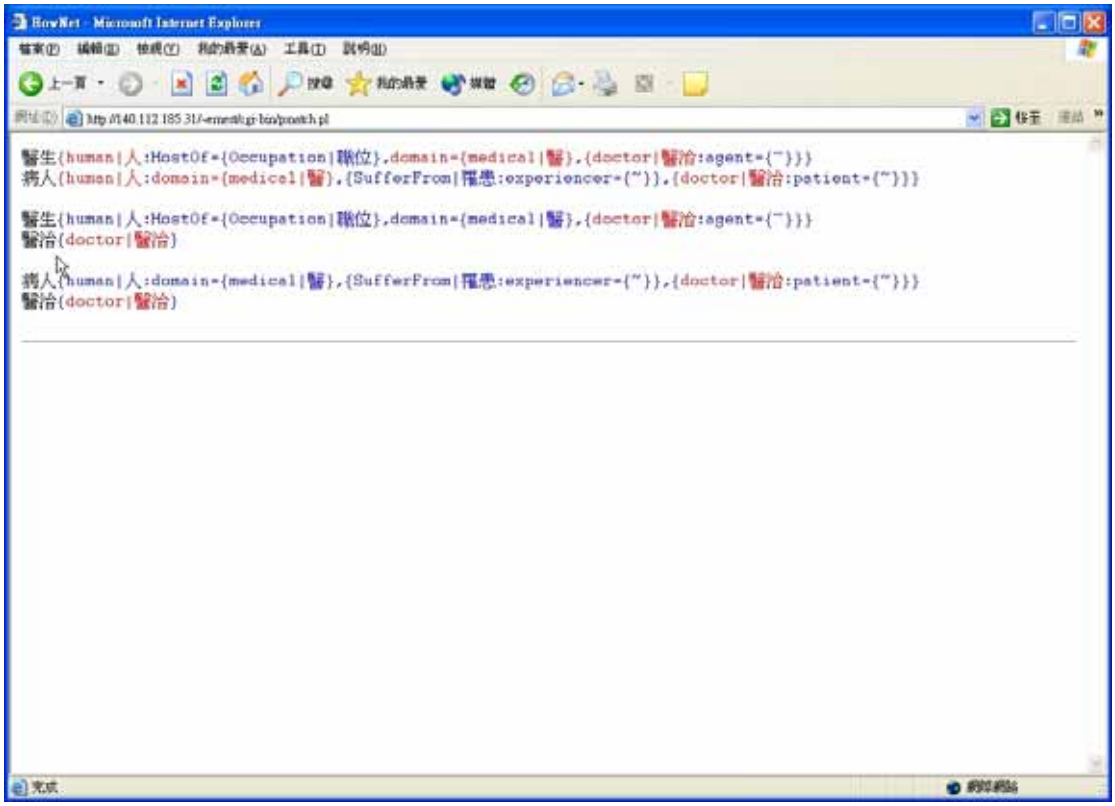


圖 8. 輸入中文用雙語辭典轉換成英文再用 Wordnet 找出詞的語意關係

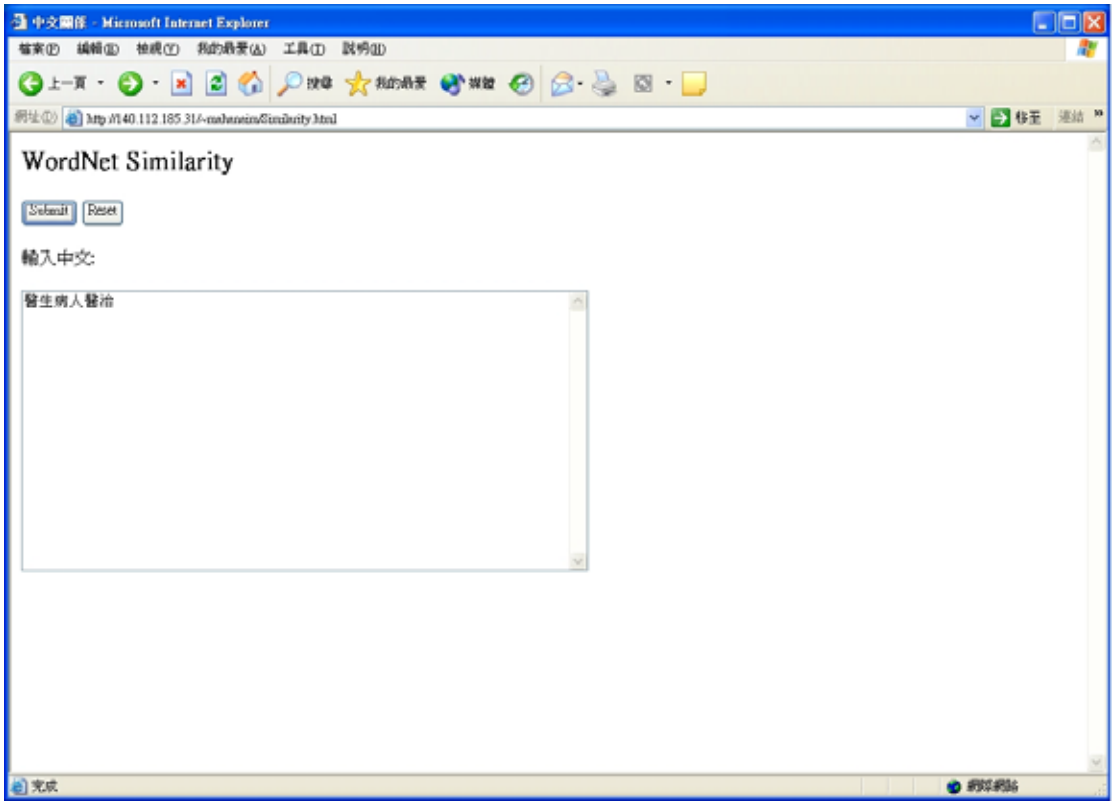
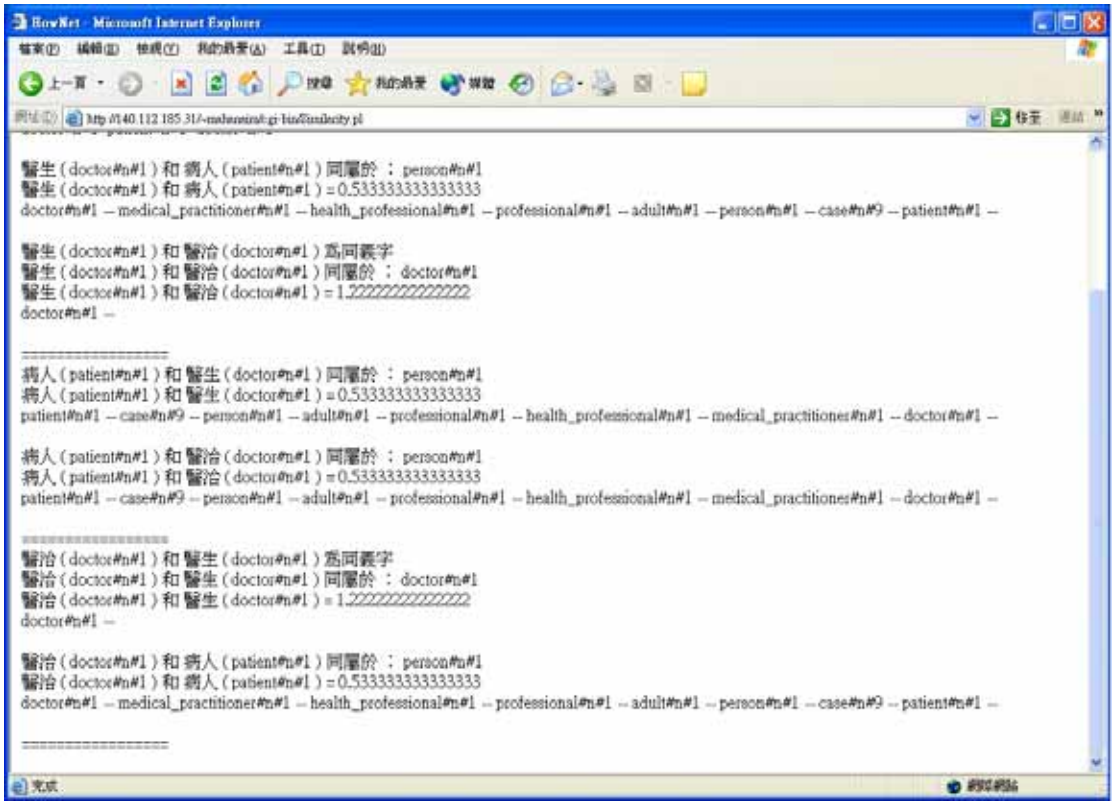


圖 9. 輸入中文用雙語辭典轉換成英文再用 Wordnet 找出詞的語意關係得到的結果



首先我們必需有兩份資訊，第一是英文 parsed 的結果，第二是中、英文各字的對應分數。

我們使用 $EN[1..length(EN)]$ 和 $CH[1..length[CH]]$ 來表示英文和中文，其中 $EN[i]$ 表示英文的第 i 個詞(單字)， $CH[i]$ 表示中文的第 i 個詞(由中文分句程式而得的詞)。

接著，我們使用 $V(en_{start},en_{end},ch_{start},ch_{end})$ 來表示當 $EN[en_{start}..en_{end}]$ 和 $CH[ch_{start}..ch_{end}]$ 對應時所得到的分數，則我們的目標就是找出在 $EN[1..length(EN)]$ 和 $CH[1..length(CH)]$ 中能使 V 最大的 en_{start} 、 en_{end} 、 ch_{start} 、 ch_{end} 。

我們首先從英文 parsed 的結果來觀查：我們使用 mlparser 來處理英文，結果爲一 syntax tree，其中每一個 leaf 都對應到該英文句子中的一個詞(單字)。因此我們的第一步驟即是要找出這些 leaf 所對應的中文。

對於每一個 leaf $EN[i]$ ，我們先找出所有 $V(i,i,j,j),1\le j\le length(CH)$ 不爲零的 j ，並將其依 V 值重新排序，假設爲 $\{j_1,j_2,...,j_n\}$ ，並令 j_k 所對應的 $V(i,i,j_k,j_k)$ 值爲 v_k 。因此可知， $v_1\ge v_2\ge...\ge v_n$ 。令 $dev=\frac{1}{n}Sigma_{k=1}^{n}(v_1-v_k)$ 。令 C 爲 $EN[i]$ 所對應中文的集合，則若 $v_k+dev\ge v_1$ 且 j_k 和 $\min C$ 或 $\max C$ 的距離不超過一定值，則我們將 j_k 加入到 C 裡。在此「定值」考慮英文的複合名詞，中文有部份機會譯爲「□□的□□」，因此給定爲 2。當 C 計算完畢，則我們可以令 $EN[i]$ 所對應的中文爲 $CH[\min(C)..\max(C)]$ ，而令 $V(i,i,\min(C),\max(C))=Sigma_{e\in C}V(i,i,e,e)$ 。