中文句子相似度之計算與應用

鄭守益　梁婷
國立交通大學資訊科學系
{gis93540, tliang}@cis.nctu.edu.tw

摘要

近年來受惠於國內外各項語料庫資源的建置及網際網路上的大量中文語料，使電腦語文輔助教材的涵蓋層面日趨廣泛。因此如何產生大量且具高品質之輔助教材日益受到許多自然語言處理研究者的重視。有鑑于此，本論文提出以中文句子相似度爲基礎的研究與應用。相似度的計算乃考慮句子的組合及聚合性。我們實作此一應用，並提出解決未知詞的語意計算問題的方法。實驗結果顯示系統的檢索 MRR 值可以提升到 0.89 且每一檢索句皆可找到可堪用之例句。

1. 緒論

句子是可完整表達語意的基本單位[21]，也是語法的具體表現。因此，在語言學習中，學童若是學會了各種句型，也就學會了隱含在句型中的語法規則。藉由語言學家的歸納整理[14]，我們知道句子的結構並不是詞語的隨意組合，而是依照一定的「語法規則」。根據[15]，語法規則可進一步分爲「組合規則」及「聚合規則」。組合規則是指語法單位的橫向組合，例如，「我」、「買」、「書」這三個詞彙可以組合成「我買書」，但卻不能組合成「書買我」。當詞組合成結構之後，將具有語法意義，並使得整體結構的意義大於個別詞彙的意義總和，例如：「綠」、「葉」這兩個詞各自有其意義，但組合之後則形成了「綠」修飾「葉」的語法意義。

至於聚合規則是指在句子中，每個位置的語法單位都有其適合替換的詞語集合，例如，在「我買書」這個句子裡，「我」可以替換成「你」，但「買」卻不能替換成「花」。句子中的聚合替換規則可以視爲詞彙的語義替換問題，例如：語義同屬植物的「花」、「草」可以互相替換。

句型在學習語法時十分重要，因此融合語法變化的「句型練習」就成爲國小學童語言學習時的一個重要活動[18]。國語習作是現行國語課程的輔助教材，主要供國小學童課後練習使用，而習作的內容中幾乎每課都有「造句」、「照樣造句」、「替換語詞」等句型的練習 [16]。然而，由於習作中所提供的例句數量不多，再加上國小學童不論在閱讀的文章數量及習得的詞彙數量皆有所不足，因此，本研究之目的爲設計一有效率之句子相似度計算方法，以自動擷取國小學童句型練習中的「照樣造句」所需的範例例句。我們將句子相似度定義爲計算兩個句子之間的語法規則相似度，也就是說如果兩個句子的語法組合及聚合規則越相似，則其相似度越高。

句子相似度計算可依照語句的分析深度分成兩種方式。一種是基於向量空間模型的方法，把句子當成詞的線性序列，因此語句相似度衡量機制只能利用句子的表層資訊，即組成句子的詞的語義資訊。由於不加任何結構分析，這種方法在計算語句之間的相似度時無法考慮句子整體結構的相似性。例如在[8] [20]是以比對相同辭彙來計算相似度，對於句子之中，普遍存在的同義或近義詞之間的取代及比對，並沒能有效的解決。在[9]則提出搭配語義詞典檢索，並分配字義權重，以解決單純語義匹配的問題；但是，只使用語義詞典檢索來作爲相似度的計算依據，而沒有考慮到句子內部的結構和詞彙之間的相互關係，因此準確率並不理想。在[11]中提出使用編輯距

離的方法，其規定的操作模式，並不完全適用於整體句義相似的計算，也缺乏同義或近義詞替換的設計。另一方面，使用統計之語言模型的方法 [6]則需要建置大量的訓練語料。在[2][4]中結合了語義詞典檢索方法及傳統編輯距離方法[10]的優點，並利用 HowNet[5]和《同義詞詞林》[7]兩種語義辭典，以計算辭彙之間的語義距離，同時賦予不同編輯操作不同的權重，因此具有較好的輸出結果。由於其方法是基於同義詞典，來進行語義判定，因而衍生出未知詞及專有名詞語義判定的問題。另外。檢討其所使用的編輯操作權重，篩選候選句的計算方式，及評估輸出結果的方法，仍有改進的空間。

另一種方法則是對語句進行結構的句法與語義分析，並在分析結果的基礎上進行相似度計算，例如[17][19]先對被比較的兩個句子進行深層的句法分析找出依存關係，並在依存分析結果的基礎上進行相似度計算，但目前的語義依存句法分析器的準確率只有 86%，因此造成依存分析的結果並不準確，導致句子的核心詞無法正確判斷，因而產生了錯誤的計算結果。

在本論文中，我們提出以聚合規則相似度和組合規則相似度來設計並實作中文相似句子擷取系統。我們使用兩個句子中所含的詞彙之同義或近義詞來計算聚合語義的相似度，以及改良式編輯距離計算的方法，並設計新的權重配置比例、候選句篩選原則。在語義計算過程中，加入詞性標記資訊，以節省語義計算的次數，最後使用語義相似度矩陣，將所輸出的參數加以正規化，以取代人工評分的方法。

由於本論文所提之「句子相似度」可應用於學童句型練習中「照樣造句」所需之範例例句，操作方法即是按照原來句子的句型造句，例如：輸入「今天看到一幅畫」，輸出「昨天想到一個人」，因此只需要計算詞的線性序列相似度，而不需用深層的樹狀結構分析。此外，本研究將同時使用全域匹配(Global Alignment)及局部匹配(Local Alignment)的策略，求取兩句在全句和部分句段的結構相似度。

2. 聚合結構相似度

我們定義句子的聚合結構相似度爲兩個句子之間的詞語是否可使用同義或者近義詞替代。例如：「我愛你」與「你喜歡哥哥」就是一對聚合規則相似的句子。本研究改良並採用，以語義爲基礎的編輯距離演算法，來計算句子之間的聚合規則相似度。重新考慮編輯操作代價，及使用上下文資訊以解決未知詞及專有名詞語義判定問題，並利用網路語料來改進因資料稀疏而無法有效進行詞義比對的計算問題。

2.1 語義相似度計算

一般的編輯距離指的是，從一個句子變爲另一個句子，所需要的最小編輯操作的步驟數。傳統的編輯操作共有「保留」、「插入」、「刪除」和「替換」四種。以下圖爲例：（ —: 代表刪除；◯ 代表保留）

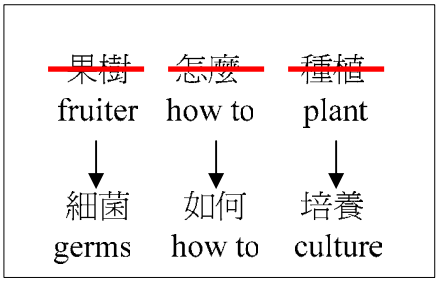

圖 1(a)：傳統編輯距離

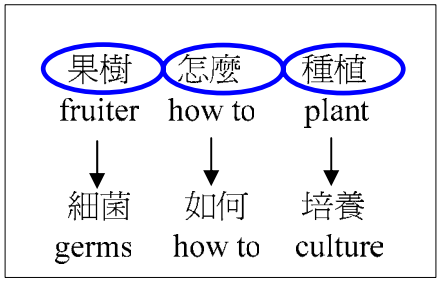

圖 1(b)：語義型編輯距離

從上面的計算過程可以看出，若僅使用編輯距離的方法，則計算出的語義距離和的實際情況將有許多差距。就語義而言，詞語之間的編輯操作代價應當各不相同。例如，上述兩句範例，雖然字面上的詞彙都不一樣，但若細細探究其中的涵義，可以發現其中的詞彙所扮演的文法角色及上位詞的語義內涵，有一定程度的相似。此外若在檢索目標的句子或短語的詞彙之中，加入具有修飾功能的詞彙，其語義也具有相似性。例如「果樹怎麼成功種植」與「細菌應如何快速培養」可視爲相似句。基於以上的觀點，本研究採用編輯距離的改進演算法[2]，即以辭彙爲基本的計算單位，同時以 HowNet 和《同義詞詞林》作爲語義距離的計算資源，以涵蓋更多的中文詞彙。

在《同義詞詞林》中，將詞彙按照語義關係的遠近親疏，賦予了一或多個語義代碼。按照樹狀的層次結構把所收錄的詞條分別歸類。同一層的詞語其關係有詞義相同或相近，或詞義在真實世界中有很強的相關性。例如：「大豆」、「毛豆」和 「黃豆」在同一層；這些詞不同義，但相關。從樹狀結構來看，《同義詞詞林》有五層結構，越靠近根節點，語義的概念越抽象。具體的詞彙，只分佈在節點末端。利用《同義詞詞林》來計算詞與詞之間的語義距離，可視爲單純的代數操作。但詞義的操作代價，應隨同義詞典的級距分歧度加大而增加，而非等量的增加。因此我們定義 X、Y 兩詞之間的詞義距離如下：

$$Dist(X,Y) = \underset{x\in X, y\in Y}{Min}\ dist(x,y) \qquad (1)$$

其中 x,y 爲分屬於 X，Y 兩詞之語義集合，根據《同義詞詞林》的結構，其計算公式定義如下：

$$dist(x,y) = \mathrm{Csim}(x,y) + (ld * \alpha) \qquad (2)$$

$$\mathrm{Csim}(x,y) = [((|n-5|*(4-n))/10) + 0.1] \qquad (3)$$

Csim(x,y)是指兩詞在同一棵語義結構樹之中，且兩詞的詞義從第 n 層結構開始有所不同；而 *ld* 爲該兩詞彙在個別的句子中的位置差距，α 爲系統定義的同義詞位移編輯代價。由於同義詞在距離相對詞語的位置超過三個以上時，其語義角色就已經產生變化，例如：「我對你很好，對不對？」句中的「對」這個詞，雖然，在句子中出現了兩次，但其語義已然不同。爲將詞語的位移控制在三以內，於是我們以計算同義詞林第一層語義代價除以三，將 α 設爲 0.3。

另外，我們認爲在詞語中進行插入或刪除等操作，將有可能影響並改變句子的整體意義及結構，因此這些操作將有較高的操作代價。我們定義爲：若進行刪除或插入操作，則操作代價應等同於兩詞不同義的代價，因此，我們以 n=0 代入公式（3）計算而設定爲 2.1。

HowNet 中同義詞的定義爲具有相同的英語譯文（W_E）和語義定義（DEF）的辭彙，其操作

代價設定爲 0.1。例如「愛」和「喜歡」，其簡化詞條如下：

表 1：HOWNET 同義詞舉例

NO	W_C	G_C	W_E	DEF
514	愛	V	love	FondOfl喜歡
89949	喜歡	V	love	FondOfl喜歡

在系統的計算過程中，先比對在 HowNet 中，兩詞是否爲同義詞，若是則兩詞之操作代價爲 0.1，若否則比對《同義詞詞林》並引用(1)作爲決定操作代價之依據。

2.2 未知詞詞義處裡

我們定義在 HowNet 及《同義詞詞林》中未收錄的詞彙稱爲未知詞。我們先在現有的語料庫中搜尋包含該未知詞的句子，並使用上下文資訊的相似度來協助判斷兩個詞語的相似程度，設定前後文的詞窗個數爲三個鄰近詞，並用共現值 I 來抽取相關度高的上下文詞組，其計算公式如下：

$$I(X_u, Z_w) = \log \frac{f(X_u, Z_w)/N}{(f(X_u)/N)(f(Z_w)/N)} \quad (4)$$

其中 N 表示語料詞數量，X_u 爲未知詞，Z_w 爲位於 X_u 前後的 3 個詞的任一詞彙，$f(X_u)$， $f(Z_w)$ 分別表示 X_u，Z_w 在語料庫中出現的次數，$f(X_u, Z_w)$ 表示詞 X_u，Z_w 一起出現的次數。經過實驗測試我們將共現值門檻值定爲 7，挑選出的關聯詞將作爲 X_u 的詞義集合。假設查詢句和目標句中分別有未知詞 X，Y，且他們的關聯詞分別是 $x_1, x_2 \cdots x_m$ 和 $y_1, y_2 \cdots y_n$ ，則同樣的我們可利用公式(1)來建立X和Y的相似矩陣M如下：

$$M(X,Y) = \begin{bmatrix} Dist(x_1, y_1), Dist(x_1, y_2) ..., Dist(x_1, y_n) \\ ..... \\ Dist(x_m, y_1), Dist(x_m, y_2) ..., Dist(x_m, y_n) \end{bmatrix} \quad (5)$$

再利用公式(6)計算出 X 和 Y 之間的語義相似度 S(X,Y)：

$$S(X, Y) = \frac{\sum_{i=1}^{m} Min[Dist(x_i, y_1), Dist(x_i, y_2), ...Dist(x_i, y_n)]}{m} \quad (6)$$

在未知詞詞義選取處理時，若無法獲得關係詞作爲語義相似度的判斷時，我們將使用網路語料作爲關係詞的查詢來源，本研究使用 Google 作爲查詢的搜索引擎，其步驟如下：

步驟 1: 查詢 Google 首頁，得知目前全部的待查網頁數量，作爲 N 值。

步驟 2: 使用未知詞 X_u 作爲搜索詞，查出 $f(X_u)$，並選取前 10 個摘要內容，作爲鄰近詞的抽取對象。

步驟 3: 將抽出的含有未知詞 X_u 的句子，進行斷詞，並進行鄰近詞抽取，詞窗個數爲三個鄰近詞。

步驟 4: 將未知詞 X_u 及鄰近詞一同作爲搜索詞，送進Google分別查出 $f(Z_w)$ 及 $f(X_u, Z_w)$。

步驟 5: 利用公式(4)，並篩選出超過門檻值的詞。

步驟 6: 將鄰近詞組代入語義相似度計算矩陣，計算關鍵詞對的語義相似度。

3. 組合結構相似度計算

如前所述，中文句子相似性的計算考量如下：

（1）任意句子中的詞組，其詞性角色的排列不可任意錯置，但可容許有限度的局部置換，例如：「我(N)吃(Vt)了(ASP)一(DET)碗(N)麵(N)」，不可寫成「我(N)一 (N)麵(N)吃(Vt)了(ASP) (DET)碗」，後者明顯不合文法；但「一(DET)碗(N)麵(N)是(V)我(N)吃(Vt)了(ASP)」，卻可以說的通。

（2）句子中的詞與詞之間，具有可插入適當空隙的特性，例如：名詞的前面應可容許插入相關的形容詞，「我(N)吃(Vt)了(ASP)一(DET)碗(N)麵(N)」，也可寫成「我(N)吃(Vt)了(ASP)一(DET)碗(N)很(Dfa)燙(VH)的(DE)麵(N)」。

由於全域匹配在計算上，將會考慮查詢句的詞性標記的整體的序列，因此可充分反映上述的第(2)項特點；而局部匹配，則只考慮由查詢句的詞性標記序列末端往前回溯的最佳子序列，因此可充分反映上述的第(1)項特點。我們將使用動態規劃演算法，分別計算句子與句子之間的，整體及局部相似度後，再依一定比例加權計算：

設A,B爲兩中文句之詞性標記序列，分別表示爲：A:{$a_1,a_2,\cdots a_n$};B:{$b_1,b_2,\cdots b_m$}，序列中之任一詞性標記 a_i和b_j，$a_i \subseteq A$，$b_j \subseteq B$，”-“爲序列中因不匹配而插入之間隙(gap)，$\sigma(a_i,b_j)$表示a_i和b_j比較時的分數值，我們定義爲：

$$\sigma(a_i,b_j)=2, \text{ for all } a_i=b_j$$

$$\sigma(a_i,b_j)= -1, \text{ for all } a_i \neq b_j$$

$$\sigma(a_i,b_j)= \sigma(-,b_j)=-1$$

我們利用 SMITH WATERMAN 所提出的全域相似匹配演算法[12]來找出匹配句，並以公式(7)計算出以詞性標記爲主的兩句相似度值，其中 l 爲兩序列比對時之最大長度。

$$\mathrm{G}Sim(\mathrm{A},\mathrm{B}) = \frac{Score(A,B)}{n+m} \tag{7}$$

$$Score(A,B) = \sum_{i=1}^{l} \sigma(\mathrm{a_i},\mathrm{b_i}) \tag{8}$$

另外一方面，我們利用改良式的 SMITH WATERMAN 算法[1]來找出局部相似的候選句。其計算匹配的路徑不需要到達矩陣的盡頭，如果某種匹配的分數值不會因爲增加匹配的數量而增加時，這種匹配就是最佳的。其相似度值爲：

$$\mathrm{L}Sim(\mathrm{A},\mathrm{B}) = \frac{Score(A,B)}{n+m} \tag{9}$$

$$Score(\mathrm{A},\mathrm{B}) = Max\{C[i,j]\} \tag{10}$$

其中 $Max\{C[i,j]\}$ 爲計算矩陣中分數最高的數值。

4. 混合式的中文句子相似度的計算應用系統

綜合上述所提的組合及聚合結構相似度計算，我們提出了一個混合式的中文句子相似度的計算應用系統(系統架構圖見圖 2)，在進行句義相似度計算時，主要分爲以下步驟：

步驟 1:利用[13]進行中文句斷詞並自動標注詞性標記。

步驟 2:同義詞擴展：爲了使候選句能更具有多樣性及提高系統的召回率，因此我們對斷詞之後的各個辭彙進行同義詞擴展。本系統使用 HowNet 語義詞典作爲詞擴展的資源。

步驟 3:候選句檢索：我們假設，如果一個候選句中所含的詞語，與查詢句的相同詞或同義詞越多，就越有可能是我們要擷取的相似句。因此，我們設定候選句的標準爲：候選句的詞數不能大於檢索句的 3 倍，而符合的詞數不得小於檢索句詞數的三分之一，並按照句子權重由大到小排序，選擇前 100 句作爲候選句。

步驟 5:句子聚合結構相似度計算

步驟 6:句子組合相似度計算

步驟 7:分別依各項不同的計算數值，取前 10 句候選句，作爲答案。

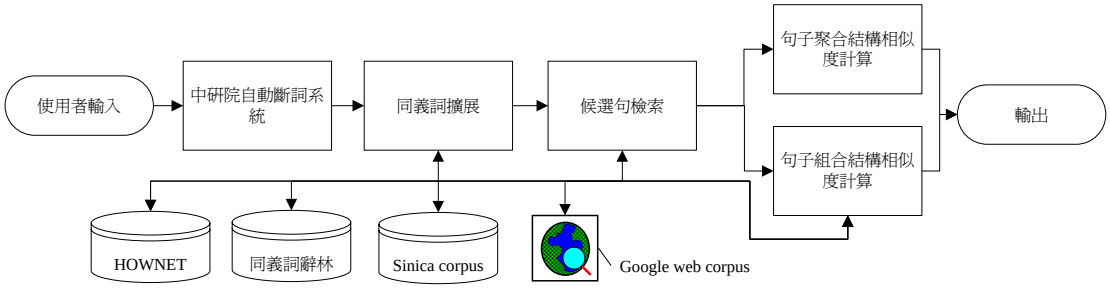

圖 2 系統架構圖

5. 實驗與分析

我們採用中央研究院平衡語料庫 3.0 版，作爲系統的候選句及查詢句的語料庫，其中包含了 500 萬已標記的中文語料。我們從中隨機選取包含 5 到 8 個詞彙的短句 100 個作爲查詢句。

本論文設計成四種不同的實驗做比較：

(1) BaseLine：以[2]中所提的詞彙作爲計算單位的動態規劃編輯演算法。

(2) M1：在詞義判斷過程中，利用語料庫的上下文資訊，來處理未知詞。

(3) M2：在語義判斷過程中，加入網路語料上下文資訊，來處理未知詞。

(4) M3：利用本論文所提的組合及聚合規則來計算相似度。

又依選擇候選句的指標不同，使用 MRR(Mean Reciprocal Rank)分別測試其對於選擇正確的候選句的影響：

(1) OC (Operation Cost): 使用原有的編輯距離作爲抽取候選句的標準。設 n,m 分別爲候選句及查詢句的長度：

$$OC = \sum_{i=1}^{n} \sum_{j=1}^{m} dist\ (x_i, y_j) \qquad (11)$$

(2) NOC (Normalized Operation Cost): 使用候選句及查詢句中，句子所含的詞數進行正

規化：

$$\mathrm{NOC} = \frac{OC}{Max(n,m)} \tag{12}$$

(3) SWR (Semantic Weight Ratio):傳統上多數句子相似度的評分標準都是以編輯距離操作代價作爲句子相似度的評分標準，但是這樣的分數會因爲句子的長度不同，而造成長句往往分數會高出許多。然而將原始的編輯操作代價進行正規化，亦無法避免因編輯距離的不同，而給予較客觀的相似度評估。因此，本論文乃設計在語義計算過程中，所產生的編輯操作代價，依照正負相關系數的門檻值，切分成正相關係數及負相關係數，再透過與原始編輯距離的計算，產生出詞語語義貢獻度SWR(Semantic Weight Ratio)。其計算方式如下：設S_q, S_t爲兩中文句詞彙序列，P爲所有編輯距離操作代價之分數總和，Q爲所有負相關係數總和[1]，則

$$\mathrm{SWR}(S_q, S_t) = \frac{P-Q}{P} \text{ ， } 0 \le \mathrm{SWR}(S_q, S_t) \le 1 \tag{13}$$

其值越接近 1 則表示S_q, S_t句中所含的相似詞語越多。

(4) PCRC (POS Construction Related Coefficient): 結合全域及局部的匹配相似度，作爲判斷候選句及查詢句之間的結構相似度，經實驗將兩項數值的比重設定如下：

$$\mathrm{PCRC} = (0.6 \times LASim(A,B)) + (0.4 \times GASim(A,B)) \tag{14}$$

(5) CSSS (Combine Semantic and Structure Similarity): 結合語義及語法結構，作爲抽取候選句的標準，因本系統主要將應用於國小學童的照樣造句的活動之上，因此將比較偏重於結構方面的相似度，因此將兩項數值的比重設定如下：

$$\mathrm{CSSS} = (0.4 \times \mathrm{SWR}) + (0.6 \times \mathrm{PCRC}) \tag{15}$$

以上的各項標準所篩選出的候選句集合，我們使用人工方式以 MRR 值來評定其效能。使用此值能測量出系統產生出第一個語義最相近的例句的平均名次。若第一個結果即爲最佳匹配，則分數爲 1，第二個匹配分數爲 0.5，第 n 個匹配分數爲 1/n，若無匹配的句子，則分數爲 0。最終的分數爲所有得分之和。另外我們還觀察各項實驗中，找不到例句的查詢句的數量變化，我們使用「NON」來表示其數值。

另外，我們在使用上下文資訊，進行語義相似度計算時，使用詞性標記資訊，以減少計算的雜訊。例如：在「張三站在椅子上」，「飛彈上有字」這兩句都有「上」(Ncd（位置詞）)。因此，在使用上下文資訊計算詞意相似度時，我們將不考慮下列詞性的詞：

Da（數量副詞）、Caa（對等連接詞）、Cbb（關聯連接詞）、Nep（指代定詞）、Neqa（數量定詞）、Nes（特指定詞）、Neu（數詞定詞）、Nf（量詞）、Ncd（位置詞）、Nd（時間詞）、Nh（代名詞）、P（介詞）、Cab（連接詞）、Cba（連接詞）、Neqb（後置數量定詞）、DE（的，之，得，地）、I（感嘆詞）、T（語助詞）、SHI（是）、V_2（有）

以下爲各個模組採用 CSSS 標準所產出的範例，查詢句爲：「世上還有癡心的人嗎？」

[1] 本研究設定之門檻值爲操作代價大於同義詞林第三層語義代價，也就是以n=2 代入公式(3)而設定爲 0.7。

表 2 查詢結果範例

查詢句	模組	排序	候選句	CSSS	MRR
世上還有癡心的人嗎？	M3	1	音樂真的有那麼深的殿堂嗎？	0.71	1
		2	你有足夠的耐性嗎？	0.68	*
		3	我還有追求幸福的權利嗎？	0.63	*
	M2	3	有這樣子的人啊？	0.58	*
		4	中國也有瓷器嗎？	0.55	0.25
		5	屈辱的生，英勇的活。	0.53	*
	M1	4	屈辱的生，英勇的活。	0.52	*
		5	中國也有瓷器嗎？	0.50	0.2
		6	唉唷，還有巧的呢！	0.49	*
	BaseLine	7	限電方式也有雙贏的？	0.45	*
		8	中國也有瓷器嗎？	0.44	0.125
		9	並且也發表您的看法嗎？	0.43	*

5.2 實驗結果與分析

圖 3 及圖 4 顯示了 MRR 值在四個實驗模組的分佈情況。從圖顯示，相對於其他的模組，M3(MRR 值平均皆大於 0.7)可以有效提昇相似候選句的選取。並且也不會因爲使用不同的篩選模組而降低候選句的品質。另一方面，實驗也顯示所設計設的 CSSS，其 MRR 值平均大於 0.68。相對於其他篩選標準，CSSS 可以控制候選句的品質，並且可以將相似句的 rank 值提升。由於 CSSS 的 MRR 值顯示了正相關，跟 PCRC 及 OC 比較起來，當同時考慮語義相似度時，它可以改善 MRR 值到達 0.89。

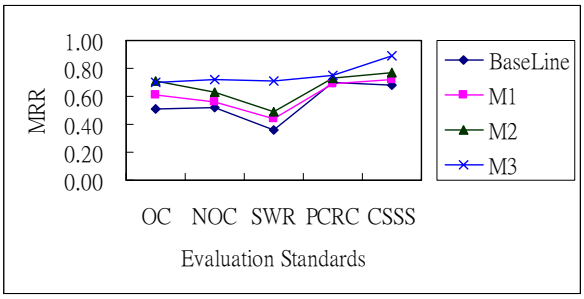

圖 3: 四個實驗模組的 MRR 值

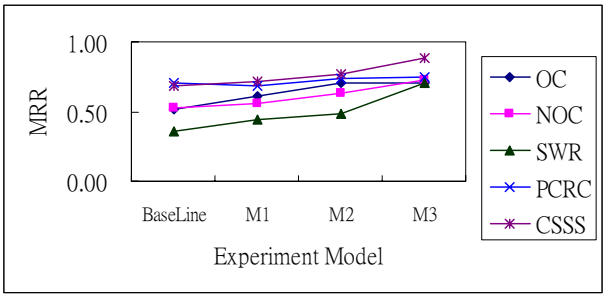

圖 4 不同系統之 MRR 值變化

另一方面，圖 5 及圖 6 顯示 NON 值，在使用不同的條件句篩選標準情況下，四個實驗模組的分佈情況。從圖顯示，M3 的 NON 平均值皆小於 2 (如果同時使用 PCRC 或 CSSS 則可以下降到 0)。這意味 M3(相對於其他模組而言，)可以更有效的抽取出相似的候選句。另一方面，PCRC 及 CSSS 的 NON 值平均小於 3.5，因此相對於其他模組而言， 在抽取候選句的時候如果同時考慮語義則可以將 NON 的值降到 0。

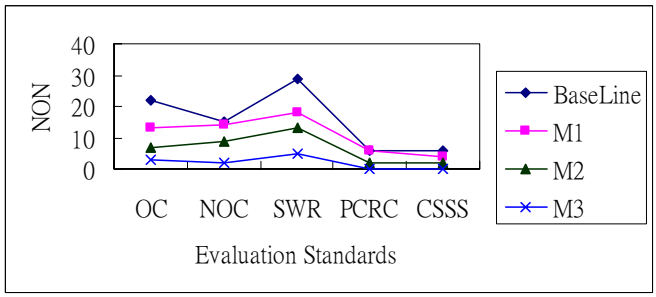

圖 5: 不同實驗指標之 NON 值變化。

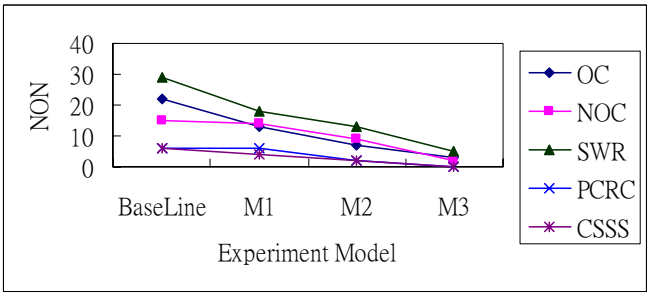

圖 6: 不同系統之 NON 值變化。

6 結語

在本論文中，我們提出新的中文句子相似度計算策略，並應用於中文習作中的範例句產生之自動化。此例句產生系統可自動從語料庫中抽取相似句以作爲學童練習造句時之參考。此系統主要有如下之改良：

(1) 改良語義計算所使用之編輯距離計算方式，加入限制同義詞或近義詞位移的操作代價，以解決詞語因重複出現，而造成語義權重判斷錯誤的問題。

(2) 使用上下文資訊之相似度，作爲判斷詞義相似的標準，以解決辭典未收錄詞彙的詞義

判斷問題。

(3) 爲了解決資訊稀疏的問題，在現有的資料庫無法提供有效判斷詞義的上下文資訊時，將採用 Web Corpus 來輔助。

(4) 使用詞性標記資訊協助判斷詞義，去除不含有效語義判斷成分的詞類，並減少相似度比對時的計算量。

(5) 使用全域相似度匹配及局部相似度匹配，並結合詞性標記，加權計算句子之間的組合結構相似度。

(6) 改良並設計新的句子相似度計算公式，結合句子的聚合及組合相似度，並可依照系統的需求，機動調整權重，以符合使用者的需求。

致謝

我們感謝中央研究院資訊科學所詞庫小組提供之線上斷詞系統 (http://ckipsvr.iis.sinica.edu.tw/)。

參考文獻

[1] Altschul, S.E., Gish, W.: Local alignment statistics, Vol. 266. Methods Enzymol (1996) 460-480

[2] Che, W. X., Liu, T., Qin, B., Li, S.: Similar Chinese Sentence Retrieval based on Improved Edit-Distance, Vol. 14(7). High Technology Letters (2004) 15-20

[3] Chen, K.J., Ma,W.Y.: "Unknown Word Extraction for Chinese Documents,"Proceedings of COLING 2002, pages 169-175

[4] Chatterjee, N.: A Statistical Approach for Similarity Measurement Between Sentences for EBMT, Proceedings of Symposium on Translation Support Systems. 2nd Indian (2001)

[5] Dong, Z. D., Dong, Q.: HowNet, http://www.keenage.com (1999)

[6] Li, S., Zhang, J., et al.: Semantic Computation in Chinese Question-Answering System. 2002, Journal of Computer Science and Technology, 17(6): 933

[7] Mei, J.J. et al.: TonYiCi CiLin－thesaurus of Chinese words (同義詞詞林), Shangwu Yinshuguan (商務印書局香港分館), Hong Kong (1984)

[8] Nirenburg, S.: Two Approaches of Matching in Example-Based Machine Translation, Proc. TMI-93, Kyoto, Japan, 1993

[9] Qin, B., Liu, T., Yang, W., Zheng, S., Li, S.: Chinese Question Answering System Based on Frequently Asked Questions, Journal of Harbin Institute of Technology May (2003)

[10] Ristad , E. S., Yianilo, P. N.: Learning string-edit distance. Vol. 20(5). IEEE PAMI (1998) 522

[11] Ristad, E. S., Yianilo, P. N.: Learning string-edit distance. 1998, IEEE PAMI, 20(5): 522

[12] Simith, T. F., Waterman, M. S.: Identification of Common Molecular subsequence, Vol. 147. Journal Mol. Biol. (1981) 195-197

[13] 中央研究院線上斷詞系統，http://ckipsvr.iis.sinica.edu.tw/

[14] 謝國平，《語言學概論》台北：三民書局，2002 年 頁 195

[15] 葉蜚聲，徐通鏘，「語言學綱要」，台北：書林，2001 年，頁 97-106。

[16] 蔡米凌，「國小三年級學童作文句型結構之分析研究—以嘉義地區爲例」，國立嘉義師範學院國民教育研究所碩士論文，1997 年。

[17] 穗志方，「語句相似度研究中的骨架依存分析法及其應用」，北京大學博士學位論文，1998 年。

[18] 陳玫秀，「學前兒童國與句型結構之分析研究」，國立師範大學特殊教育研究所碩士論文，1990 年。

[19] 李彬，劉挺，秦兵，李生，「基於語義依存的漢語句子相似度計算」，電腦應用研究，2003 年。

[20] 我國簡易刑事判決的製作輔助系統 (Decision support for criminal summary

judgment)，第七屆人工智慧與應用研討會論文集 (TAAI'02)，178-183。台中，台灣，15 November 2002 年。

[21] 胡百華，「華語的句法」，台北：阿爾泰，1984 年。