以範例爲基礎之英漢 TIMSS 試題輔助翻譯

張智傑　　劉昭麟
國立政治大學 資訊科學系
{ g9512 ,chaolin }@cs.nccu.edu.tw

摘要

本論文應用以範例爲基礎的機器翻譯技術，應用英漢雙語對應的結構輔助英漢單句語料的翻譯。翻譯範例是運用一種特殊的結構，此結構包含來源句的剖析樹、目標句的字串、以及目標句和來源句詞彙對應關係。將翻譯範例建立資料庫，以提供來源句作詞序交換的依據，最後透過字典翻譯，以及利用統計式中英詞彙對列和語言模型來選詞，產生建議的翻譯。我們是以 2003 年國際數學與科學教育成就趨勢調查測驗試題爲主要翻譯的對象，以期提升翻譯的一致性和效率。以 NIST 和 BLEU 的評比方式，來評估和比較線上翻譯系統和本系統所達成的翻譯品質。

關鍵詞：自然語言處理，試題翻譯，機器翻譯，TIMSS

1. 緒論

國際教育學習成就調查委員會(The International Association for the Evaluation of Education Achievement, 以下簡稱 IEA)[20]主要目的在於了解各國學生數學及科學(含物理、化學、生物、及地球科學)方面學習成就、教育環境等，影響學生的因素，找出關聯性，並在國際間相互作比較。自 1970 年起開始第一次國際數學與科學教育成就調查後，世界各國逐漸對國際數學與科學教育成就研究感到興趣，IEA 便在 1995 年開始每四年辦理國際數學與科學教育成就研究一次，稱爲國際數學與科學教育成就趨勢調查(Trends in International Mathematics and Science Study，以下簡稱 TIMSS)，至今已辦理過 1995、1999、2003 和 2007 共四屆，共有 38 個國家參加。

我國於 1999 年開始加入 TIMSS 後，由國科會委託國立台灣師範大學科學教育中心(以下簡稱師大科教中心)負責試題翻譯及測驗工作。1999 年的調查對象只有國中二年級學生， 2003 年的調查對象包括四年級及八年級學生。翻譯試題主要的流程包含：從 IEA 取得試題內容，由師大科教中心決議進行翻譯工作分配、中文試題交換審稿校正及翻譯問題討論，最後將中文翻譯試題定稿。至目前爲止，師大科教中心已將 1999 和 2003 年試題內容和評量結果，公布於台灣 TIMSS 官方網站[21]，以提供研究之參考。在 TIMSS 的試題內容上，主要的題型種類有選擇題和問答題，試題句型大多爲直述句和問句結構所組成，選擇題則多了誘答選項。

以往使用人工翻譯雖然可以達到很高的翻譯品質，但是需要耗費相當多的人力資源和時間，而且在翻譯過程中不同的翻譯者會有不同的翻譯標準(例如：相同的句子，翻譯後的結果不同)；相同的翻譯者也可能在文章前後翻譯方式不一致而產生語意上的混淆。因此間接影響試題難易程度。若直接將英文詞彙透過英漢字典翻譯成相對的中文詞

彙，翻譯的結果可能會不符合一般人的用詞順序。另外中文的自由度較高，很容易造成翻譯上用詞順序的不同。例如：“下圖顯示某一個國家所種穀物的分布圖”，也可翻譯爲“某一個國家所種穀物的分布圖，如下圖顯示”。可能會影響到受測者的思緒，使作答時粗心的情形會增加。因此，若能利用機器翻譯(machine translation)的技術來輔助翻譯以及調整詞序，以期提高翻譯的品質和效率。

在人工智慧領域，機器翻譯是一個很困難的問題。機器翻譯是指將一種自然語言經過電腦運算翻譯成另一種語言，困難程度也跟來源句和目標句有關，像是英文和葡萄牙文語言的特性較相近，較容易翻譯。而中文跟英文詞序差異很大，且中文比較沒有特定的語法，寫法較自由，對翻譯來說較爲困難。機器翻譯發展至今已經超過 50 年。Dorr 等學者[9]將現在機器翻譯依據系統處理的方式來分類，分成以語言學爲基翻譯(linguistic-based paradigms)，例如基於知識(knowledge-based)和基於規則(rule-based)等；以及非語言學爲基翻譯(non-linguistic-based paradigms) ，例如基於統計(statistical-based)和基於範例(example-based)等。

以知識爲基礎的機器翻譯(knowledge-based machine translation)系統是運用字典、文法規則或是語言學家的知識來幫助翻譯。Knight 等學者[11]結合 Longman 字典、WordNet 和 Collins 雙語字典建立一個知識庫，運用在西班牙文翻譯成英文。這種利用字典來幫助翻譯的系統，會有一字多義的情形發生，一個詞彙在字典中通常有一個以上的翻譯。以英翻中爲例“current”這個字在字典裡就有十多種不同的翻譯，即使專家也無法找出一個統一的規則，在何種情況下要用何種翻譯，所以在翻譯的品質和正確性上很難滿足使用者。因此，翻譯系統通常都會限定領域來減少一字多義，例如 current 在電子電機類的文章中出現，最常被翻譯爲電流，在文學類的文章中，最常被翻譯爲現代。

統計式機器翻譯(statistical machine translation，以下簡稱 SMT)是將語料在翻譯之前就已經過計算轉換成統計數據，不需要在翻譯過程中作龐大的數學運算，能有較高的效能。Brown 等學者[6]於 1990 年以英文及法文的雙語語料爲來源，提出統計式雙語翻譯架構。假設目標語言爲 T 及來源語言爲 S，P(T)爲目標語言 T 在語料庫中出現的機率，稱爲語言模型(language model)，P(S|T)爲目標語言 T 翻譯成來源語言 S 的機率，稱爲翻譯模型(translation model)。SMT 系統需要大量的語料庫輔助，大多都需要具備雙語對應的語料庫(parallel corpora 或稱 bilingual corpora)，再透過機率公式計算出機率模型。其中 SMT 困難的地方在於需要收集大量可用的雙語語料，當語料越多建立模型所花費的時間越多。Oct 等學者[16]提出單字式(word-based)翻譯模型運用在詞彙對準(word alignment)，並且發展出 GIZA++這套系統。Koehn 等學者[12]進一步將單字式轉變成片語式(phrase-based)翻譯模型，運用片語式翻譯模型翻譯的結果會比單字式翻譯的結果要正確。

以範例爲基礎的機器翻譯(example-based machine translation，以下簡稱爲 EBMT)的相關研究已有相當多年歷史，在 1990 年日本學者 Sato 和 Nagao[19]所提出的 EBMT 是將翻譯過程分爲分解(decomposition)、轉換(transfer)和合成(composition)三步驟。分解階段是將來源句到範例庫中搜尋，並將所搜尋到 word-dependency tree 當作來源句的 word-dependency tree，並且形成來源句的表示式；轉換階段將來源句的表示式轉換成目

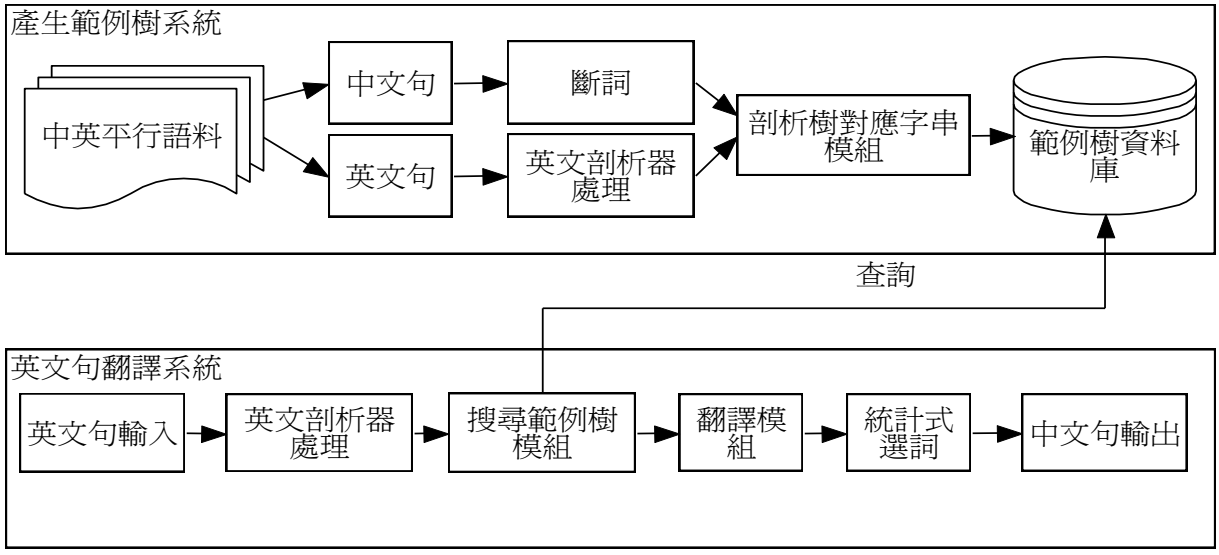
標句的表示式；合成階段將目標句的表示式展開爲目標句的 word-dependency tree，並且輸出翻譯結果。Al-Adhaileh 等學者[5]將 structured string tree correspondence(SSTC) [7]運用在英文翻譯成馬來西亞文上，SSTC 是一種能將英文對應馬來西亞文的結構，但此結構並沒有解決詞序交換的問題。目前較完整的 EBMT 系統有 Liu 等學者所提出 tree-string correspondence (TSC)結構和統計式模型所組成的 EBMT 系統[13]，在比對 TSC 結構的機制是計算來源句剖析樹和 TSC 比對的分數，產生翻譯的是由來源詞彙翻譯成目標詞彙的機率和目標句的語言模型所組成。

黃輝等學者所提出的 translation corresponding tree (TCT) [24]，TCT 是針對英文翻譯成葡萄牙文的系統，在 TCT 結構上可以記錄來源句詞彙和目標句詞彙對應的關係、來源句詞彙和目標句詞彙對應的翻譯結果和詞序，但是 TCT 是二元的剖析樹，也就是每個節點都只有兩顆子樹，在 TCT 上詞序只用布林值(boolean value)來記錄，所以 TCT 只能運用在二元剖析樹上。但是有些剖析器所產生剖析樹是多元樹，因此我們提出雙語樹對應字串的結構(bilingual structured string tree correspondence，簡稱爲 BSSTC)可以運用在多元剖析樹上，並且 BSSTC 可在翻譯過程中當作詞序交換的參考，根據我們實驗結果，我們能有效的調動詞序，以提升翻譯的品質。完成詞序交換後，再透過字典翻譯成中文，最後運用統計式選詞模型，產生了初步翻譯結果，但本系統尚屬於半自動翻譯系統，故需要人工加以修飾編輯。

除了本節簡單介紹本研究以外，我們將在第二節描述整個系統的架構，第三節說明本篇論文所運用的技術，第四節則呈現出我們的實驗結果，第五節則是結論。

2. 系統架構

由於我們的目的在於利用中英互爲翻譯的句子找出詞序關係，並且將英文句和中文句詞序的資訊儲存在電腦中，儲存的格式是將中英文句的詞序關係記錄在英文剖析樹的結構中，此結構將成爲之後英文句的結構調整爲適合中文的結構的參考。最後再將英文詞彙翻譯成中文詞彙，並利用統計式選詞選出最有可能翻譯成的中文詞彙，讓翻譯的結果更符合一般人的用詞和順序。



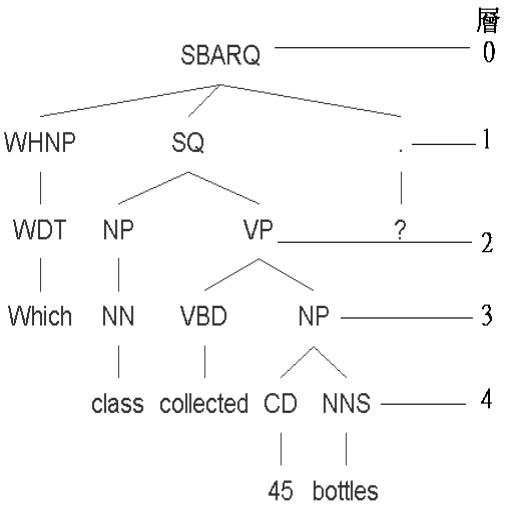
圖一、系統架構圖

本系統的架構如圖一所示。我們針對範例樹產生系統和英文句翻譯系統這兩部份分別簡介如下。

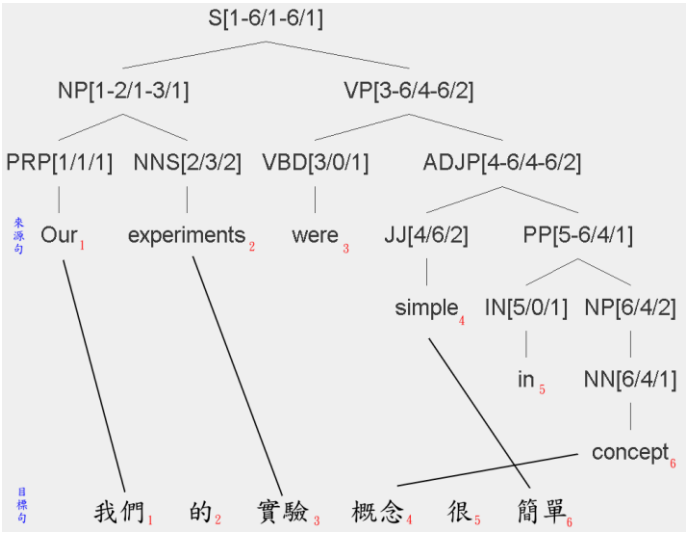
- **範例樹產生系統：** 這個系統利用中英平行語料，這裡的中英平行語料必需要一句英文句對應一句中文句，且每一組中英文句都要是互爲翻譯的句子。中文句經過斷詞處理後，被斷成數個中文詞彙，以空白隔開；英文句經過英文剖析器建成英文剖析樹。將斷詞後的結果和英文剖析樹經過剖析樹對應字串模組處理，建成英文剖析樹對應字串的結構樹，此結構樹稱爲範例樹。再將每個範例樹取出子樹，並且判斷是否有詞序交換，將需要詞序交換的範例樹全部存入範例樹資料庫中方便搜尋。
- **英文句翻譯系統：**當輸入英文句後，先將句子透過英文剖析器，建成英文剖析樹。有了英文剖析樹就可以透過搜尋範例樹模組，標記英文剖析樹上需要調動詞序的結構，並依照所標記的詞序作調整。詞序調整完成後再將英文結構樹中的英文單字或片語透過翻譯模組做翻譯。其中翻譯模組包含了大小寫轉換、斷詞處理、stop word filtering及stemming，之後將處理過的詞彙透過字典檔做翻譯[3]。每個英文單字或片語都可能有一個以上的中文翻譯，因此需要選詞的機制來產生初步翻譯結果，此翻譯結果尚需要人工作後續的編修。

3. 系統相關技術

根據上一節系統架構的描述分爲範例樹產生系統和英文句翻譯系統兩大系統。範例產生樹系統的執行流程爲先將中文句斷詞和剖析英文句，再將斷詞和剖析後的結果輸入至剖析樹對應字串模組，並將處理後的範例樹存入資料庫中。英文句翻譯系統的執行流程區分爲三大部分，第一部分是由搜尋範例樹模組，將英文剖析樹跟範例樹資料庫作比對，並且將未比對到的子樹做修剪；第二部分將修剪後的剖析樹輸入到翻譯模組翻成中文；第三部分以中英詞彙對列工具及 bi-gram 語言模型，計算出中英詞彙間最有可能之翻譯組合。



圖二、英文剖析樹



圖三、BSSTC 結構的表示法

3.1 雙語樹對應字串的結構(BSSTC)

在建立 BSSTC 結構之前，我們必須將中英平行語料中的中英文句先作前處理，我們將英文句透過 StanfordLexParser-1.6[17]建成剖析樹，剖析樹的每個葉子節點爲一個英文單字，並以英文單字爲單位由 1 開始標號。這裡我們將樹根定義爲第 0 層，樹根的子樹是第 1 層，越往下層數越大，故葉子節點必定是英文單字，且不屬於任何一層，如圖二所示。而中文句是使用中研院 CKIP 斷詞系統[1]作斷詞，並以斷詞後的單位由 1 開始標號。這裡的中文句代表來源句；英文句代表目標句。本結構是假設在中英文對應都是在詞彙的對應或連續字串的對應。

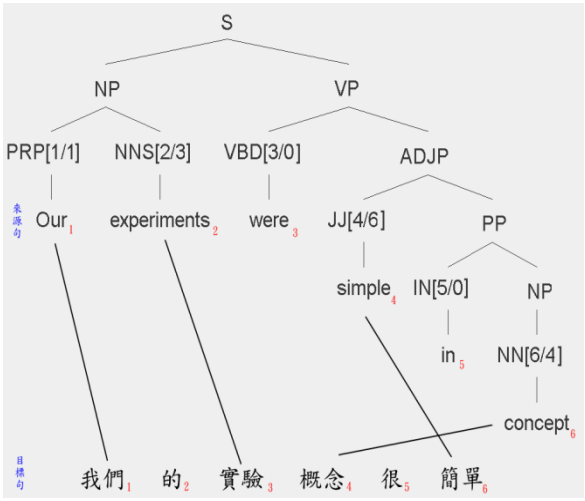
假設剖析樹的節點集合 N={N_1, N_2, …, N_m}，m 爲剖析樹上節點個數，對任一節點 $n \in N$，n 有三個參數分別是 n[STREE//]、n[/STC/]和 n[//ORDER]；我們以 n[STREE/STC/ORDER]來表示。爲了方便說明，若節點 n 只有 n[STREE//]和 n[/STC/]，則以 n[STREE/STC/]表示。再假設 $n_{C(n)}$爲節點 n 有 1 到 C(n)個子節點。n[STREE//]爲節點 n 所涵蓋來源句的範圍，層數最大節點的 n[STREE//]必定對應到一個來源句單字，此參數的功用爲當作每個節點的鍵值(primary key)，故在同一棵剖析樹中 n[STREE//]不會重複。n[/STC/]表示以 n 爲樹根的子樹，所涵蓋來源句字串的範圍對應到目標句字串的範圍；n[/STC/]也可以是一個數字，表示此子樹包含的目標句字串爲目標句字串中的一個字；n[/STC/]也可能是 0，代表來源句無法對應到目標句。n[//ORDER]是由 n[/STC/]計算出來，n[//ORDER]是用來表示來源句跟目標句詞序對應的關係，若來源句跟目標句有詞序不同的情形，就可由 n 與所有兄弟節點的 n[//ORDER]來判斷。ORDER 的範圍由 1 到 C(n)，當 ORDER 越小，代表 n 所對應目標句範圍，比其他兄弟節點的目標句範圍更靠近句子的前段。

圖三是一個 BSSTC 結構的例子，來源句爲英文：“Our experiments were simple in concept”；目標句爲中文：“我們的實驗概念很簡單”。首先英文句必須先建成剖析樹，每個葉子節點爲一個英文單字，並以英文單字爲單位做標號，例如：“Our(1)”, “experiments(2)”, “were(3)”, “simple(4)”, “in(5)”,“concept(6)”。另外中文句經過斷詞的處理後，以斷詞後的單位做標號，例如：“我們(1)”, “的(2)”, “實驗(3)”, “概念(4)”, “很(5)”, “簡單(6)”。中英對應句都標號後，以標號爲單位開始做詞彙對準(word alignment)，並標記在剖析樹的節點上。剖析樹是用文法結構來分層，不同層節點能對應到不同的範圍的目標句字串。n[STREE/STC/]若爲 VP[3-6/4-6/]，則 STREE 代表節點 VP 對應來源句第三到第六個字 “were simple in concept“；STC 代表“were simple in concept”對應目標句的第四到第六個字“概念很簡單”。$n_{C(n)}$[STREE/STC/ORDER]的兄弟節點(sibling node)若爲 JJ[4/6/2]和 PP[5-6/4/1]，我們可以觀察到 JJ 的 ORDER 大於 PP 的 ORDER，故 PP[5-6/4/1]的中文對應「概念」在 JJ[4/6/2] 的中文對應「簡單」之前。

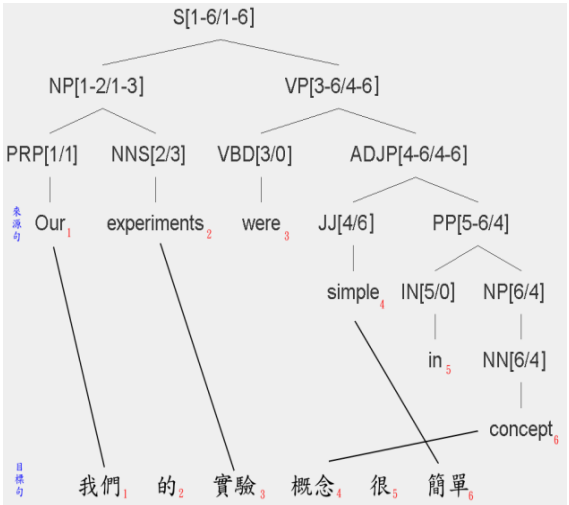
3.2 建立 BSSTC 結構和產生範例樹

建立 BSSTC 結構必需要有英文跟中文互爲翻譯的句子，建構的順序是從最底層也就是層數最大的開始標記，再一層一層往上建置到第 0 層爲止，標記參數順序是先將所有節點的 n[STREE//]和 n[/STC/]標記完後，再標記 n[//ORDER]。首先，標記最底層 n[STREE//]

的方法，是將最底層的節點 n 所對應葉子節點的編號標記在 n[STREE//]。如圖三節點 NNS 所對應來源句的“experiments”的編號爲 2，故 NNS[STREE//]中的 STREE 標記爲 2。接著標記最底層 n[/STC/]的方法是尋找中英對應句中互爲翻譯的中文詞彙和英文詞彙，也就是詞彙對準。詞彙對準若採用人工方式，則相當耗時費力，其本身也是一項困難的研究。因此，我們在此用一個簡單的方法，首先先將中文句經過斷詞處理，這裡我們使用中研院 CKIP 斷詞系統[1]；將英文句每個英文字查尋字典檔，查尋後可能會有超過一個的中文翻譯，將這些中文翻譯跟斷詞後的中文詞彙一個一個作比對，如有比對到則認定互爲翻譯，並且標記 n[/STC/]在剖析樹上。如圖三來源句的“experiments”在字典中的翻譯有“實驗”、“經驗”和“試驗”，將這三個中文翻譯到目標句去比對，此例子將會比對到目標句第三個詞彙“實驗”，接著將目標句“實驗”的編號標記在 NNS[2/STC/]中的 STC 上。最後將比對到的個數除以英文句單字的個數，稱爲對應率。最佳情況下是每個英文單字都有相對應的中文翻譯，對應率爲 1；最差的情況下每個英文單字都沒有相對應的中文翻譯，對應率爲 0，所以對應率會落在 0 到 1 之間，值越大代表對應率越高。我們需要夠大的對應率，才能認定爲範例樹。因此，需要定一個門檻值來篩選，根據實驗結果當門檻值越高留下來的範例樹越少，而門檻值越低會使翻譯的品質下降。



圖四、僅標記最底層



圖五、僅標記 STREE 及 STC

目前範例樹只將最底層的 n[STREE/STC/]標記完成，如圖四，現在要逐層將未標記 n[STREE/STC/]的節點標記上去。n[STREE//]標記的方式，是將 n_1 到 $n_{C(n)}$的 STREE 都加入 ES 中。ES 爲用來儲存 $n_{C(n)}$ [STREE//]中 STREE 的集合。當層數越小，則 n[STREE//]將會涵蓋 1 個以上的來源句的詞彙。若 $n_{C(n)}$ [STREE//]爲一個範圍，則將此範圍最大和最小的值加入 ES，最後 ES 內可能爲一個數字或兩個以上的數字這兩種情況，如只有一個數字則 n[STREE//]只標記該數字，如有兩個以上的數字則 ES 中最小和最大的數值標記在 n[STREE//]上，格式爲 n[最小-最大//] ；n[/STC/]標記的方式，是將 n_1 到 $n_{C(n)}$的 STC 都加入 CS 中。CS 爲用來儲存 $n_{C(n)}$ [/STC/]中 STC 的集合。當層數越小，則 n[/STC/]將會涵蓋 1 個以上目標句的詞彙。如 $n_{C(n)}$ [/STC/]爲一個範圍，則將此範圍最大和最小的值加入 CS。若 $n_{C(n)}$ [/STC/]出現 0 則不加入 CS。最後 CS 可能爲空、一個數字或兩個以上的數字這三種情況，如爲空則將 n[/STC/]標記爲 0，若只有一個數字則 n[/STC/]只

標記該數字，假如有兩個以上的數字將 CS 中最小和最大的 STC 標記在 *n*[/STC/]上，格式爲 *n*[/最小-最大/]。

假如我們現在要標記圖五第一層的節點 VP，則必需將節點 VP 的子節點 VBD 和 ADJP 的 VBD[3//]及 ADJP[4-6//]中的 STREE 加入 ES 中，因此 ES 包含了 3、4 和 6 三個數字，所以 VP[STREE//]中的 STREE 標記爲 3-6。接著標記 STC，將節點 VP 的子節點 VBD 和 ADJP 的 VBD[3/0/]及 ADJP[4-6/4-6/]中的 STC 加入 CS 中，因爲 0 不會被加入 CS 中，因此 CS 只有 4 和 6 兩個數字，所以 VP[3-6/STC/]中的 STC 標記爲 4-6。

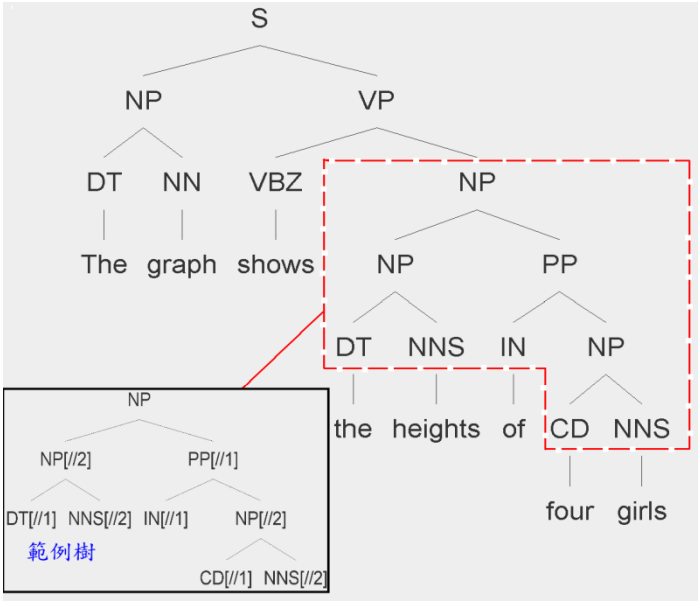
最後，整棵剖析樹的 STREE 跟 STC 都已經標記完成，如圖五，只剩下 ORDER 還沒標記上。ORDER 處理方式分爲兩部分，第一部份：STC 爲 0 之兄弟節點按照由左至右的順序編號；第二部分：比較 *n* 與 STC 非 0 之兄弟節點的大小，並接在第一部份的編號後，由小到大繼續標記編號。例如圖五若要標記 JJ[4/6/ORDER]和 PP[5-6/4/ORDER]的 ORDER，則將 JJ[4/STC/]中的 STC=6 和 PP[5-6/STC/]中的 STC=4 由小排到大，所以 PP[5-6/4/ORDER]中的 ORDER 標記爲 1，JJ[4/6/ ORDER] 中的 ORDER 標記爲 2。

利用上述的方法得到範例樹，如圖三。如直接用整個句子的範例樹到資料庫中作搜尋，將很難搜尋到相同的範例樹，因爲句子越長句子的結構會越複雜，所以相同結構的句子重複出現的可能很低。因此，我們將範例樹的所有子樹分別取出來，每一個子樹所包含的範圍的都是英文句的子句，在不同的句子裡可能會有相同結構的子句，不但可以增加比對到的機率，也能增加範例樹的數量。最後記錄在範例樹資料庫的內容，只有範例樹和 ORDER 參數。STREE 和 STC 不需記錄的原因是每一個句子的每個詞彙都在不同的位置上，則在資料庫中不需要記錄 STREE 和 STC。

範例樹的結構有可能相同，而詞序不同。例如“NP(NP(NN fork))(PP(IN of)(NP(DT the)(NN road)))”，中文翻譯爲“岔路”，而“NP(NP(NN leader))(PP(IN of)(NP(DT a)(NN company)))”， 中文翻譯爲“一間公司的領導者”。很明顯後者中英文用詞順序不同。這裡我們採用多數決，將出現過相同範例樹結構的每種詞序作統計，在範例樹資料庫中記錄出現最多次詞序的結構。如出現最多次的次數相同，則以隨機方式選擇一種記錄在範例樹資料庫中。最後再將範例樹資料庫中沒有詞序交換的範例樹刪除，只保留有詞序交換的範例樹，可以減少搜尋相同範例樹的時間。

3.3 搜尋相同範例樹

範例樹資料庫裡，每一筆資料都包含範例樹和範例樹的 ORDER，而範例樹就是用來當作調整詞序的參考。將輸入的英文句，先透過 StanfordLexParser-1.6[17]建立剖析樹，再將剖析樹中去掉葉子節點的結構，到範例樹資料庫去搜尋是否有相同結構的範例樹，這裡我們將所搜尋到相同的範例樹稱爲匹配子樹。如圖六所示，紅色虛線框是一棵子樹其結構爲“(NP(NP(DT)(NNS))(PP(IN)(NP(CD)(NNS))))”，方形框爲範例樹資料庫中其中一棵範例樹結構爲“(NP(NP[//2](DT[//1]) (NNS[//2])) (PP[//1](IN[//1]) (NP[//2](CD[//1]) (NNS[//2]))))”，我們可以發現範例樹去除 ORDER 後的結構，會跟子樹的結構完全相同，故將此範例樹認定爲匹配子樹。



圖六、剖析樹與範例樹的對應關係

根據搜尋範例樹演算法的流程，如圖七。首先將來源句的剖析樹加到佇列(queue)裡，從佇列裡面取出一棵剖析樹到範例樹資料庫中，搜尋是否有相同結構的範例樹；如爲否，則將此棵樹的下一層的子樹加入佇列，加入佇列的順序爲左子樹到右子樹；如爲是，則將該樹的 ORDER 標記在來源句的剖析樹上，繼續取出佇列內的剖析樹，直到佇列裡沒有剖析樹爲止。所以來源句的剖析樹是由一個以上的匹配子樹所組成。

輸入：來源句剖析樹 S
　　範例樹資料庫 D={D_1, D_2, ..., D_m}，$D_i \in D$
　　D_i 包含 T_i 與 O_i ，i 爲 1 到 m
　　T_i 是第 i 棵範例樹，O_i 是 T_i 所標記的詞序
開始
　　設佇列 Q 用來儲存剖析樹，初始爲 NULL
　　S 加入 Q
　　當 Q≠NULL
　　　從 Q 中 pop 一棵範例樹
　　　如果在 D 中搜尋到相同的範例樹 T_i
　　　　則將 O_i 標記在 S 上
　　　否則將下一層子樹加入 Q
結束
輸出：標記好 ORDER 的剖析樹

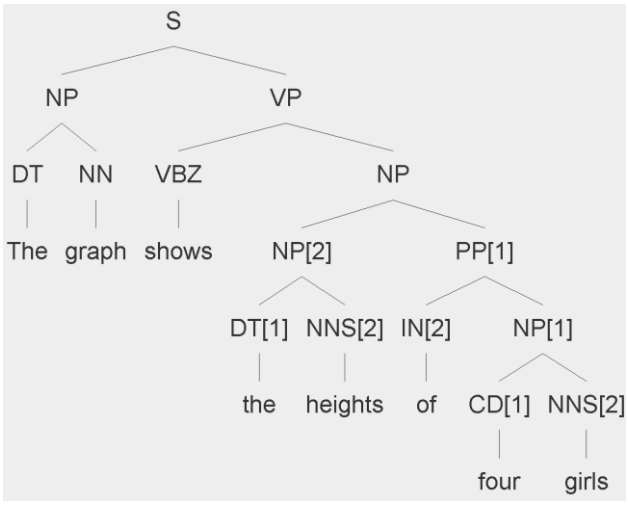
圖七、搜尋範例樹演算法

圖六爲剖析樹搜尋範例樹的情形。來源句：“The graph shows the heights of four girls”，剖析樹爲“(S(NP(DT The)(NN graph))(VP(VBZ shows)(NP(NP(DT the)(NNS

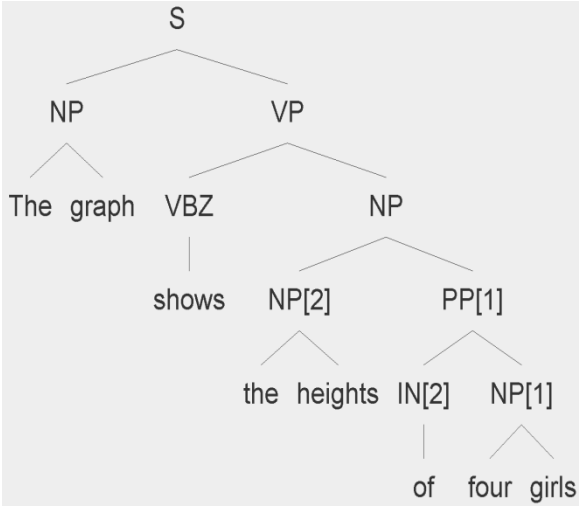
heights))(PP(IN of)(NP(CD four)(NNS girls)))))”。透過搜尋範例樹演算法找出匹配子樹，首先以節點 S 爲樹根的剖析樹到資料庫作搜尋，搜尋時不包含葉子節點，此例子沒搜尋到匹配子樹，則將節點 S 的子樹 NP 和 VP 加入佇列中。接下來將從佇列中取出的子樹爲 NP，到範例樹資料庫搜尋匹配子樹，但資料庫中沒有相同的範例樹，此時 NP 的子樹皆爲葉子節點，所以並無子樹在加入佇列中。依照先進先出的原則下一個從佇列取出的是 S 的右子樹 VP，在範例樹資料庫中還是搜尋不到，因此要將 VP 的子樹 VBZ 和 NP 加入佇列中，但 VBZ 爲葉子節點，故只有 NP 加入佇列中。接下來是子樹 NP 從佇列中被取出來，子樹 NP 在資料庫中搜尋到相同的範例樹，如圖六的範例樹就是所搜尋到的匹配子樹，因此將範例樹的 ORDER 標記上去，標記後的剖析樹將如圖八所示。此時佇列中已經爲空，搜尋範例樹的流程到此爲止。

標記完 ORDER 之後，將沒有標記的子樹作修剪，也就是將不用作詞序交換的子樹修剪到最小層樹。如圖八節點 S 的右子樹、NP[2]和 NP[1]的子樹皆不需要作詞序交換，因此修剪的結果爲“(S(NP The graph)(VP(VBZ shows)(NP(NP[2] the heights)(PP[1](IN[2] of)(NP[1] four girls))))) ”，如圖九所示。最後從層數最大的每個兄弟節點開始逐層往上依照優先權順序調整剖析樹的結構；調整後的結果將會輸入到翻譯模組產生翻譯。若我們直接取來源句剖析樹的葉子節點作翻譯，將會成爲單字式的翻譯，我們將無法對詞組或片語作翻譯。翻譯的部分會在下一節會作詳細說明。

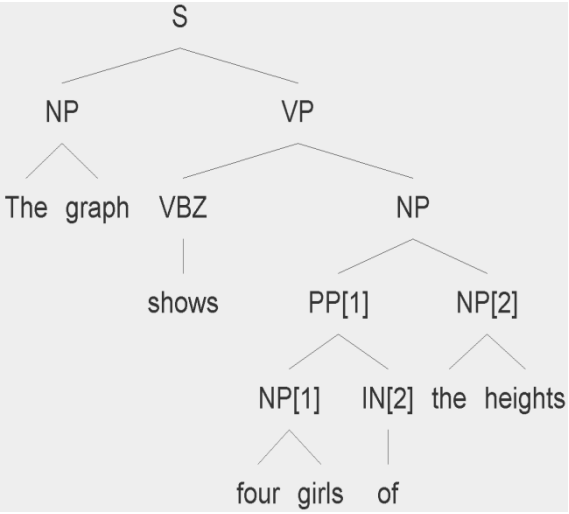
圖九的剖析樹有四層，首先將第四層的兄弟節點“(IN[2] of)(NP[1] four girls)”，依照 ORDER 的順序調整後的順序爲“(NP[1] four girls) (IN[2] of) ”，接下來第三層的兄弟節點“(NP [2] the heights)(PP[1] (NP[1] four girls)(IN[2] of))”交換後的順序爲“(PP[1] (NP[1] four girls)(IN[2] of)) (NP [2] the heights)，此例子接下來詞序沒有再調動，如圖十所示；最後輸入翻譯模組的順序爲“The graph”、“shows”、“four girls”、“of”、“the heights”，由此順序分別作翻譯處理。



圖八、完成 ORDER 標記



圖九、剖析樹修剪後的結果



圖十、調整詞序後的結果

3.4 翻譯處理

經過上一節處理最後得到修剪樹，修剪樹的葉子節點可能爲英文單字(word)、詞組(term)。詞組即爲數個單字結合的字串，不一定爲完整的句子，如“would be left on the floor”或片語(phrase，如名詞片語、動詞片語、形容詞片語等) ，如“in order to”。在翻譯處理上會遇到英文單字或詞組，在英文單字的部分，直接查尋字典檔作翻譯；詞組的部分利用規則詞典檔的片語，和詞組進行字串比對，以找出符合的片語及中文翻譯。以下爲字典檔及規則詞典檔分項說明。

字典檔：字典檔部分我們使用 Concise Oxford English Dictionary[8](牛津現代英漢雙解詞典，收錄 39429 個詞彙)，將前處理過後的英文單字或片語做翻譯對等字搜尋的動作，找出所有和該英文單字的中文詞組，作爲翻譯的候選名單。如無法在字典檔中搜尋到對應的中文翻譯。如姓名和專有名詞，則直接輸出該英文字。

規則詞典檔：爲常用的名詞片語、動詞片語、形容詞片語等詞組，以及試題翻譯小組所決議之統一翻譯詞組以人工的方式建立的中英翻譯對照檔，如 in order to(爲了)。

分成單字和詞組翻譯是因爲若在規則詞典檔比對不到，則用空白來做一般字和字之間的斷詞，也就變成單字的翻譯，因爲詞組較能完整表現出動作或敘述。如只用單字作翻譯，會造成翻譯上的錯誤。惟須注意的是比對的句型若有相似結構但不同長度的字串樣式，則取長度最長的爲結果。如一英文句子爲“…as shown in diagram…”，同時滿足規則詞典檔內的“as shown in diagram”和“in diagram”片語句型，則我們會選擇長度較長的“as shown in diagram”而不是選擇“in diagram”加上“as show”作爲斷詞的結果。

在英文翻譯成中文的過程中，有些英文單字不需要翻譯或是無意義的情形，所以我們將這些單字過濾不翻譯，這些單字稱爲 stop word。例如：冠詞 the 直接去除。介系詞 for、to、of 等，若前一單字爲 what、how、who、when、why 等疑問詞，則允以刪除，另外，to 出現在句首直接刪除。助動詞 do、does 等，判斷方式與介系詞相同。

在翻譯過程中還可能出現詞幹變化(如~ing、~ed 等)和詞性變化(如動詞 break，其過去式爲 broke，被動式爲 broken，以及名詞單複數型態)。詞幹變化的部份，我們利用 Porter[22]演算法還原各詞性(名詞、動詞、形容詞、副詞)；詞性變化的部分，有些是不規則的變化，較難用演算法處理。因此，我們透過 MXPOST[14]詞性標記工具將單字加入標記，再利用 WordNet[23]依照詞性做字典檔搜尋找到原始的型態。

3.5 統計式模組選詞

本系統將英文詞彙利用上一節介紹的翻譯方式，查詢詞典找出所有可能適合英文詞彙的翻譯結果，再利用統計式模組找出最有可能的中文詞彙，此部分已經有呂明欣等學者從事這一項研究工作[3]。以下爲我們修改後的機率模型。

$$\underset{C_{1,n}}{\operatorname{argmax}} \Pr\left(C_{1,n}\middle|E_{1,n}\right) = \underset{C_{1,n}}{\operatorname{argmax}} \prod_{i=1}^{n} [\Pr(E_i|C_i)\Pr(C_i|C_{i-1})] \quad (1)$$

公式(1)中定義 C 爲中文翻譯詞彙，E 爲英文詞彙， $E_{1,n}$爲英文句有 1 到 n 個英文詞彙，中文翻譯詞彙也會有 1 到 n 個，即 $C_{1,n}$。從公式中可發現中文詞彙翻譯成英文詞彙的機率，稱爲中英詞彙對列，即$\Pr(E_i|C_i)$；以及利用前一個中文翻譯選詞的結果 C_{i-1}，找出目前中文翻譯詞彙 C_i共同出現的機率，稱爲 bi-gram 語言模型，即$\Pr(C_i|C_{i-1})$，將兩者相乘取計算後最大的機率值，以近似$\Pr(C_{1,n}|E_{1,n})$的機率值，作爲所選擇的中文翻譯詞彙。在選詞的過程中，$\Pr(E_i|C_i)$與$\Pr(C_i|C_{i-1})$的機率值皆有可能爲 0，我們將乘 0 換成乘上一個極小數(我們預設爲 10^{-6}) ，爲了避免機率值爲 0 的情形，會影響選詞的結果。以下將針對中英詞彙對列和 bi-gram 模型詳細介紹。

中英詞彙對列：將中英語料雙語語料，經過人工的中英語句對列(sentence alignment)技術，接著將中文語料利用中研院 CKIP 斷詞系統[1]加以斷詞；英文語料則是經過大小寫轉換及利用字和字之間空白斷詞，最後輸入至 GIZA++[16]及 mkcls[15]等工具，產生中英詞彙對列結果以及中英詞彙對照機率表。

bi-gram 語言模型：將中文語料統計各中文詞彙和下一個中文詞彙出現的次數，計算其出現機率。我們是利用 SRI Speech Technology and Research Laboratory 所開發的自然語言工具 SRILM[18]來建立 bi-gram 語言模型。

4. 系統翻譯效果評估

本節主要介紹利用本系統翻譯國際數學與科學教育成就趨勢調查 2003 年考題，簡稱 TIMSS2003，並將試題依照年齡別和科目別，分別比較翻譯的品質。最後將與線上翻譯以及呂明欣等學者研發的翻譯系統作比較。評估方式爲利用 BLEU 及 NIST 指標。

4.1 實驗來源

我們主要用來翻譯的來源爲 TIMSS2003 試題，區分數學及科學類別，並且以四年級及八年級爲考試對象，共有四種試題分別爲四年級數學領域 31 題；四年級科學領域 70 題；八年級數學領域 41 題；八年級科學領域 38 題。所有試題都有英文原文試題和師大科教中心所翻譯的中文試題。

所有實驗語料句對數、中英詞彙數、中英總詞彙個數及平均句長，皆如表一所示。用來建立範例樹的來源有教育部委託宜蘭縣建置語文學習領域國中教科書補充資料題庫[4] (以下簡稱國中補充資料題庫)及科學人雜誌。國中補充資料題庫以人工方式完成中英語句對列(sentence alignment)，再經過範例樹的篩選門檻值爲 0.6 的情況下有 565 句。

用來訓練選詞機率模型的來源有自由時報中英對照讀新聞及科學人雜誌。自由時報中英對照讀新聞從 2005 年 2 月 14 日至 2007 年 10 月 31 日，而自由時報中英對照讀新聞本身就已經作好中英語句對列。科學人雜誌是從 2002 年 3 月創刊號至 2006 年 12 月共 110 篇爲語料來源。

表一、實驗語料來源統計

語料	語言	句對數	辭彙數	總詞彙個數(tokens)	平均句長
國中補充資料題庫	中文	2059 句	2333	12460	6.1
	英文		2887	13170	6.4
科學人	中文	4247 句	9279	70411	16.6
	英文		10504	68434	16.1
自由時報中英對照讀新聞	中文	4248 句	19188	145336	34.2
	英文		25782	133123	31.3

4.2 實驗設計

首先，將 TIMSS2003 試題問句以逗號、問號或驚嘆號做爲斷句的單位，每個誘答選項做爲斷句的單位，若一道題目爲一句試題問句及四項誘答選項所組成，則一道題目可斷出五句。經過人工斷句處理 TIMSS2003 試題，四年級數學領域有 165 句；四年級科學領域有 262 句；八年級數學領域有 439 句；八年級科學領域有 236 句，並整理爲文字檔。翻譯時中文試題所運用的中文斷詞爲中研院 CKIP 斷詞系統[1]，英文試題所運用的剖析器爲 StanfordLexParser-1.6[17]，建立範例樹資料庫所使用的語料爲國中補充資料題庫，訓練機率模型所使用的語料自由時報中英對照讀新聞加上科學人雜誌，其中訓練語言模型得到的 bi-gram 共有 134435 個；GIZA++產生中英詞彙對列結果有 128551 組。

表二、TIMSS 試題實驗組別表[†]

八年級 2003 M 組	八年級 2003 S 組	四年級 2003 M 組	四年級 2003 S 組	八年級 2003 MS 組	四年級 2003 MS 組
TIMSS2003 國中數學領域試題	TIMSS2003 國中科學領域試題	TIMSS2003 國小數學領域試題	TIMSS2003 國小科學領域試題	TIMSS2003 國中數學及科學領域試題	TIMSS200 國小數學及科學領域試題

我們評估所使用的工具爲依照 BLEU 及 NIST 標準的 mteval-10，並且我們將參考的中文標準翻譯和系統建議翻譯，每個中文字跟中文字之間用空白作分隔，計算出各別 n-gram 及累加各個 n-gram 的 BLEU 及 NIST 值。主要評估的對象有 Google 線上翻譯、Yahoo!線上翻譯、呂明欣學者的系統(Lu)及本系統互相做比較，並且評估翻譯系統在不同年級的試題內容上，翻譯品質是否會按照越低年級其翻譯品質越好的趨勢。因此，我們將實驗組別分爲八年級和四年級；數學領域以 M 爲代號；科學領域以 S 爲代號，當

[†]本篇論文 TIMSS 試題實驗組，僅包含 2003 年試題，與呂明欣學者的實驗組並不相同。

作實驗組別的名稱。可以 TIMSS2003 分爲八年級 2003 M 組、八年級 2003 S 組、四年級 2003 M 組及以四年級 2003 S 組四組；在加上 TIMSS 2003 數學及科學領域之八年級試題，和 TIMSS 2003 數學及科學領域之四年級試題，分別爲八年級 2003 MS 組及四年級 2003 MS 組，總共六組，如表二所示。

4.3 實驗結果

依照上一節的實驗設計，我們針對 TIMSS2003 試題驗證本系統、Lu 系統及線上翻譯系統在 BLEU 和 NIST 比較數據。從表三是以 cumulative n-gram scoring 之 4-gram 爲平均值，整理之各組 NIST 及 BLEU 值之比較表。NIST 跟 BLEU 最大的不同在於，NIST 將各 n-gram 詞彙中共現（co-occurrence）的次數比的累加值，當作各 n-gram 平均資訊量的大小，而 BLEU 針對各 n-gram 匹配正確率及相似度進行計分。由此可知當參考翻譯句子和系統翻譯句子用的詞彙相同時，NIST 分數會比較高；當參考翻譯句子和系統翻譯句子用的詞彙順序較相近時，BLEU 分數會比較高。

表三、本系統、Lu 系統及線上翻譯系統之 NIST 及 BLEU 值比較表

組別	八年級 2003 M 組		八年級 2003 S 組		四年級 2003 M 組	
指標	NIST	BLEU	NIST	BLEU	NIST	BLEU
本系統	4.7002	0.1440	4.4089	0.1254	3.9819	0.1304
Lu	3.6185	0.1007	3.5831	0.0890	3.3319	0.0983
Google	4.5268	0.1467	4.8587	0.1848	3.7573	0.1016
Yahoo!	4.8793	0.1455	4.6136	0.1396	4.0457	0.1419
組別	四年級 2003 S 組		八年級 2003 MS 組		四年級 2003 MS 組	
指標	NIST	BLEU	NIST	BLEU	NIST	BLEU
本系統	4.2228	0.1018	4.8613	0.1309	4.4400	0.1138
Lu	3.2495	0.0682	3.8031	0.0966	3.4970	0.0803
Google	4.4445	0.1527	4.9343	0.1611	4.4720	0.1344
Yahoo!	4.4361	0.1442	5.0755	0.1435	4.6070	0.1436

從表三可觀察到，八年級 2003 M 組 NIST 分數以 Yahoo!最高分，但 BLEU 分數與本系統差不多，可知 Yahoo!對八年級 2003 M 組所翻譯的詞彙跟參考翻譯較相同，但 Yahoo!和本系統翻譯後詞序的正確性是差不多的。四年級 2003 M 組試題中有較多特殊符號，例如○和●等，Yahoo!及 Google 線上翻譯系統會將這些特殊符號處理成亂碼，但本系統可以將特殊符號保留下來，故四年級和八年級 2003 M 組與最高分系統的差距較小。先前我們假設翻譯品質是否會按照越低年級其翻譯品質越好的趨勢，觀察八年級 2003MS 組及小四 MS 組，可發現與假設相反，各系統在八年級 2003 MS 組的表現都比四年級 2003 MS 組要好。可推測出本系統其中一種語料爲國中補充資料題庫較符合 TIMSS 八年級 2003 的試題。

我們將八年級 2003M 組和八年級 2003S 組作比較，四年級 2003 M 組和四年級 2003 S 組作比較，可以發現各系統除了 Google 之外，在 M 組上表現都比 S 組好，因爲 M 組的試題內容包含較多的數字，對於翻譯系統較容易處理，而 S 組則包含較多專有名詞，對於翻譯系統較爲困難。接著將本系統與 Lu 系統作比較，Lu 系統和本系統的差別爲沒有作詞序的交換。經過詞序交換後，得到正確的中文詞序，因此選詞的正確性相對會提升，所以本系統在各組的表現都比 Lu 系統要好，顯示詞序交換後會得到品質較好的中文翻譯。

5. 結論

本論文提出 BSSTC 結構，此結構能夠記錄來源句詞彙的位置、目標句詞彙的位置及來源句與目標句詞彙對應的關係；並且將 BSSTC 結構運用在我們實作的翻譯系統上。本系統是利用 BSSTC 結構建立範例樹，將來源句經過搜尋範例樹演算法，來達到修正詞序的目的。最後，在依據修正後的詞序進行翻譯，翻譯時再利用中英詞彙對列工具及 bi-gram 語言模型，選出最適合的中文翻譯，產生建議的翻譯，此翻譯還需要人工編修。

TIMSS 的試題爲數學及科學類，應該要用大量數學及科學類的語料，但實際上我們並無法找到夠多的數學及科學類語料，尤其以中英對應的語料最少，所以我們選用新聞及國中補充資料題庫來擬補語料的不足。不過訓練量還算是不足夠，在選詞上會有許多機率爲 0 的情況，造成選詞錯誤。未來將盡量找尋相關領域的語料，來建立範例樹和訓練語言模型，就能針對不同領域的來客製化翻譯，使翻譯的結果更爲精確。

訓練語料中的斷詞是使用中研院 CKIP 系統，而我們翻譯使用的字典爲牛津字典，兩者所使用的字典並不相同，會使斷詞後的詞彙可能無法在牛津字典中找到，造成選詞錯誤。未來可將翻譯後的詞彙，找出同義詞來擴充詞彙數，便能增加被找到的可能性。

英文的語言特性上並沒有量詞，而中文句中運用了很多的量詞，如缺少量詞也會使中文的流暢度下將。本系統的翻譯結果也缺少中文的量詞。未來若能將翻譯結果填補上缺少的量詞，便可達到更好的品質。

致謝

本研究承蒙國科會研究計畫 NSC-95-2221-E-004-013-MY2 的部分補助謹此致謝。我們感謝匿名評審對於本文初稿的各項指正與指導，雖然我們已經在從事相關的部分研究議題，不過限於篇幅因此不能在本文中全面交代相關細節。

參考文獻

[1] 中研院中文剖析器檢索系統, http://parser.iis.sinica.edu.tw/ [Accessed: Jun. 30, 2008].

[2] 自由時報中英對照讀新聞, http://www.libertytimes.com.tw/2008/new/jan/15/english.htm [Accessed: Jun. 30, 2008].

[3] 呂明欣，*電腦輔助試題翻譯：以國際數學與科學教育成就趨勢調查爲例*，國立政治大學資訊科學所，碩士論文， 2007。

[4] 教育部委託宜蘭縣發展九年一貫課程件至語文學習領域（英語）國中教科書補充資料暨題庫建置計畫, http://140.111.66.37/english/ [Accessed: Jun. 30, 2008].

[5] M. H. Al-Adhaileh, T. E. Kong and Y. Zaharin, “A synchronization structure of SSTC and its applications in machine translation”, *Proceedings of the International Conference on Computational Linguistics -2002 Post-Conference Workshop on Machine Translation in Asia*, 1–8, 2002.

[6] P. F. Brown, J. Cocke, S. A. D. Pietra, V. J. D. Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer and P. S. Roossin, “A Statistical Approach to Machine Translation”, *Computational Linguistics*, 79-85, 1990.

[7] C. Boitet and Y. Zaharin, “Representation trees and string-tree correspondences”, *Pro-*

ceedings of the Twelfth International Conference on Computational Linguistics, 59–64, 1998.

[8] Concise Oxford English Dictionary, http://stardict.sourceforge.net/Dictionaries_zh_TW.php [Accessed: Jun. 30, 2008].

[9] B. J. Dorr, P. W. Jordan and J. W. Benoit, "A Survey of Current Paradigms in Machine Translation" *Advances in Computers*, London: Academic Press, 1-68, 1999.

[10] Google Translate http://www.google.com/translate_t [Accessed: Jun. 30, 2008].

[11] K. Knight and S. K. Luk, "Building a large-scale knowledge base for machine translation", *Proceedings of the Twelfth National Conference on Artificial intelligence*, 773-778, 1994.

[12] P. Koehn, F. J. Och and D. Marcu, "Statistical phrase-based translation", *Proceedings of the Human Language Technology Conference*, 127–133, 2003.

[13] Z. Liu, H. Wang and H. Wu, "Example-based Machine Translation Based on TSC and Statistical Generation", *Proceedings of the Tenth Machine Translation Summit*, 25–32, 2005.

[14] MXPOST, http://www.inf.ed.ac.uk/resources/nlp/local_doc/MXPOST.html [Accessed: Jun. 30, 2008].

[15] F. J. Och, "An Efficient Method for Determining Bilingual Word Classes", *Proceedings of European Chapter of the Association for Computational Linguistics*, 71–76, 1999.

[16] F. J. Och and H. Ney, "Improved Statistical Alignment Models", *Proceedings of the Thirty-eighth Annual Meeting of the Association for Computational Linguistics*, 440–447, 2000.

[17] The Stanford Parser: A statistical parser, http://nlp.stanford.edu/software/ [Accessed: Jun. 30, 2008].

[18] A. Stolcke. SRILM – an extensible language modeling toolkit. *Proceedings of the intelligence Conference on Spoken Language Processing*, 901–904, 2002. http://www.speech.sri.com/projects/srilm/ [Accessed: Jun. 30, 2008].

[19] S. Sato and M. Nagao, Toward Memory-Based Translation", *Proceedings of International Conference on Computational Linguistics*, 247–252, 1990.

[20] The International Association for the Evaluation of Education Achievement, http://www.iea.nl/ [Accessed: Jun. 30, 2008].

[21] TIMSS 中文版官方網頁，http://timss.sec.ntnu.edu.tw/timss2007/news.asp [Accessed: Jun. 30, 2008].

[22] The Porter Stemming Algorithm, http://www.tartarus.org/martin/PorterStemmer/ [Accessed: Jun. 30, 2008].

[23] WordNet API, http://nlp.stanford.edu/nlp/javadoc/wn/ [Accessed: Jun. 30, 2008].

[24] F. Wong, M. Dong and D. Hu, Machine Translation Based on Translation Corresponding Tree Structure, *Tsinghua Science & Technology*, 25–31, 2006.

[25] YAHOO! 雅虎線上翻譯，http://tw.search.yahoo.com/language/ [Accessed: Jun. 30, 2008].