淺談機器翻譯之瓶頸及目前的研發趨勢

姚念祖　蘇克毅

致遠科技股份有限公司

摘要：機器翻譯是個很吸引人的研究題材，但是卻因為自然語言中的歧義和語法不合設定問題，而造成研發上的障礙。近年來，由於語料庫資源的快速發展，在機器翻譯系統的研究領域中，可從雙語語料庫內自動抽取知識的參數式系統，逐漸取代了傳統的規則庫系統，成為研發的主流。參數式系統不但可以擺脫規則庫系統在研發和效能上的瓶頸，更可用來進行成本較低的非教導式學習，並針對各專業領域進行領域調適。我們期待結合語言學的參數式系統，能在未來達成機器翻譯實用化的理想。

(一) 簡介

所謂的機器翻譯 (Machine Translation)，指的是使用電腦，將以一種原語言 (Source Language) 書寫的文件，轉換為另外一種目標語 (Target Language)。自 1940 年代後期開始 [Hutchins 86]，機器翻譯一直是人工智慧領域的重要研發項目。這主要是因為語言向來被認為是人與動物重要的差異所在，因此能否以電腦進行如翻譯等複雜的語言處理，一直是人工智慧學門中相當引人入勝的課題。而且翻譯本身即為具有潛力的商業區塊，國際交流的興盛，更擴大了對翻譯的需求。如果能在品質方面有所突破，在專業領域的翻譯上取代人工譯者，可以預見會有相當大的市場。除此之外，機器翻譯牽涉到自然語言 (Natural Language，如中文、英文等，用以區別人造的程式語言) 的分析、轉換與生成，差不多已涵蓋了自然語言處理的所有技術，且測試方式較為明確具體 [NIST 05]，可以作為自然語言處理技術研究的研發平台。因此之故，機器翻譯多年來一直吸引著工業界投入相關之研發工作。

但是，機器翻譯若要在翻譯市場佔有一席之地，就必須面對人工譯者的競爭。由於機器翻譯的成品需以人工潤飾和審核，這部分的人力成本將會占實際運作成本的大部分。也就是譯後人工潤飾和人工直接翻譯相比，能夠節省的時間必須多到某種程度，機器翻譯才能達到實用化的階段。如果電腦的翻譯成品中仍有相當程度之誤譯，負責潤飾的人員就必須花費大量的時間，先閱讀原文了解文意，再對照機器翻譯稿，分辨正確和錯誤的翻譯，而後才能開始進行校正工作，因而大幅增加機器翻譯的成本。所以一個正確率為 70%的翻譯系統，其價值可能不及一個正確率 90%翻譯系統的一半。這就好比在採礦時，決定礦脈是否值得開採，不只是看礦物本身的價值，還要考量探礦和採礦的成本是否過高。因此在理想情況下，應讓譯後潤飾者盡量無須參照原文，即可了解文意，僅須對機譯稿作辭句上的修飾即可，就像是老師在改作文一樣。

由於有人工翻譯這項競爭方案，因此機器翻譯若要在市場上佔有一席之地，其翻譯品質必須超過一相當高的臨界點，精確度也會面臨嚴格考驗。然而因為下文中提到的種種因素，要產生高品質的翻譯並不是件容易的事，連帶使得機器翻譯的研發和實用化遇到障礙。

在下一節中，我們將先簡單介紹一般機器翻譯的作法，然後敘述研發過程中遭遇的困難。接下來說明可能的解決方式。最後一節中，則會陳述機器翻譯研究的未來與展望。

(二) 基本流程

機器翻譯系統雖然可概分為直接式 (Direct)、轉換式 (Transfer) 及中介語 (Interlingua) 三類 [Hutchins 92]，但考量實作上的困難度，目前大部分的機器翻譯系統，都是採用轉換式的做法。轉換式機器翻譯的過程，如下圖所示，可以大致分為三個部分：分析、轉換和生成。

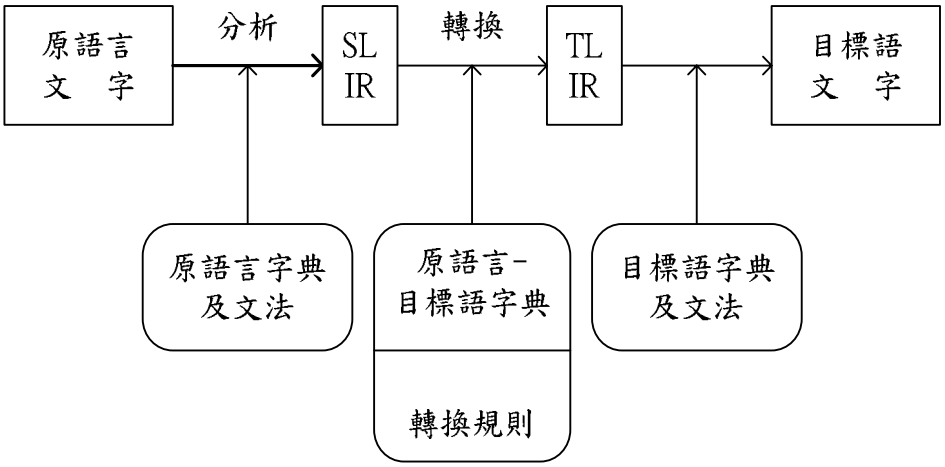
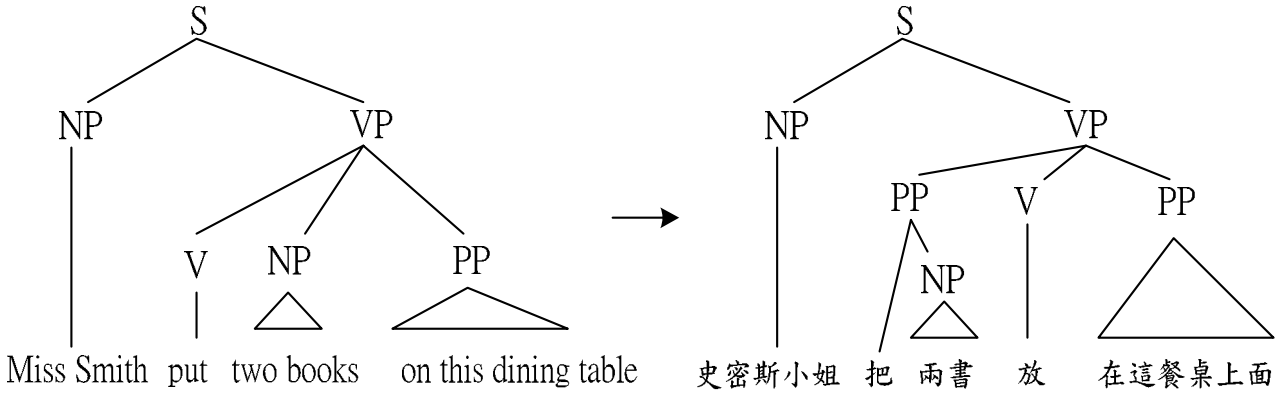


圖1. 轉換式機器翻譯流程

以「Miss Smith put two books on this dining table.」這句話的英譯中為例，首先我們會對這句話進行構詞 (morphological) 和語法的分析，得到下圖左側的英語語法樹。到了轉換階段，除了進行兩種語言間詞彙的轉換 (如「put」被轉換成「放」)，還會進行語法的轉換，因此原語言的語法樹就會被轉換為目標語的語法樹，如下方右圖所示。



語法樹的結構經過更動後，已經排列出正確的中文語序。但是直接把整棵樹的各節點排列起來，便成為「史密斯小姐把兩書放在這餐桌上面」。這其實並不是合乎中文文法的句子。因此在生成階段，我們還要再加上中文獨有的其他元素 (例如量詞「本」和「張」)，來修飾這個句子。這樣我們就可以得到正確的中文翻譯：「史密斯小姐把這兩本書放在這張餐桌上面」。為了清楚示意，以上流程僅為經過高度簡化的程序。在實際的運作中，往往需要經過多層的處理，詳細步驟請參閱 [Su 02]。

(三) 問題

自然語言處理最大的難處，在於自然語言本身相當複雜，會不停變遷，常有新詞及新的用法加入，而且例外繁多。因此機器翻譯遇到的主要問題，可以歸納為兩大項：(1) 文句中歧義 (Ambiguity)，和 (2) 語法不合設定 (Ill-formedness) 的現象。自然語言的語法和語意中，不時會出現歧義和不明確之處，需依靠其他的資訊加以判斷。這些所謂的「其他資訊」，有些來自上下文 (包括同一個句子或前後的句子)，也有些是來自是閱讀文字的人之間共有的背景知識。以下將分別說明這兩項問題。

- 歧義

所謂歧義，就是一個句子可以有許多不同的可能解釋。很多時候我們對歧義的出現渾然不覺。例如「The farmer's wife sold the cow because she needed money.」這個句子，一般人都可以正確指出此處的「she」代表的是「wife」，但是在句法上，「she」指的也可能是「cow」。雖然人類依照常識能判斷出正確的句意，但是對於依照文法規則來理解句子的電腦來說，這是一個含有歧義的句子。

在分析句子時，幾乎在每一個層次上 (如斷詞、句法分析、語意分析等)，都有可能出現歧義 [Su 02]。單字的解釋往往會因前後的文字而異。此外，判斷句子真義時需要的線索，也可能來自不同的範圍。下面這三個句子在單字的字義判斷上雖有歧義，但僅依靠句子的其他部分，即可得到進行判斷所需的充分資訊：

(1) Please turn on the light.
(2) Please turn the light on.
(3) Please turn the light on the table to the right direction.

第一句和第二句很明顯，句中的動詞就是可分動詞片語 turn on，因此我們可以輕易判斷出第二句句末的 on 是動詞片語 turn on 的一部分。但是在開頭與第二句完全相同的第三句中，同樣位置的 on 卻是介系詞片語 on the table 的一部分，與 turn 完全無關。由此可知，一個字在句子中扮演的角色，必須要參考完整的資訊後才能確定。

但有的句子若是抽離上下文單獨來看，則無法判定確切的句意。例如下面兩個句子：

(1) 他這個人誰都不相信。
(2) I saw the boy in the park with a telescope.

第一個句子，說的究竟是「他這個人不相信任何人」，還是「任何人都不相信他」？第二個句子，說的究竟是「我用望遠鏡看到一個男孩在公園裡」、「我看到一個男孩帶著望遠鏡在公園裡」、「我在公園裡用望遠鏡看到一個男孩」還是「我在公園裡看到一個男孩帶著望遠鏡」？若是沒有上下文的資訊，應該沒有人可以確定。

還有些句子，甚至需要用到文章當中沒有明言的資訊。它們雖然沒有形諸文字，但讀者仍然可依循背景知識，察知文句應有的涵義。以下面這兩個句子為例：

(1) The mother with babies under four is .....
(2) The mother with babies under forty is .....

兩個句子的句法完全相同，差別僅有「four」和「forty」一字。但是讀者卻可輕易的了解，第一個句子的「four」是用來修飾「baby」，而第二個句子的「forty」是用來修飾「mother」。讀者之所以能下意識地判斷出正確答案，憑藉的不只是文字的字面意義和語法，還要再加上生活在人類社會中的常識，了解「baby」和「mother」的合理年齡範圍。而這種「常識」，正好就是電腦最欠缺，也最難學會的部分。

我們在徵求譯者時，通常會要求譯者對稿件涉及的專業領域擁有一定的素養，為的就是避免在這種「常識」問題上出錯。這並不是一本專有名詞字典可以解決的。就像上面所舉的例子一樣，字典並不會列出四歲以下的人不可能是母親，那是讀者早該知道的。機器翻譯勢必要面對的難題之一，就是如何讓電腦得到或學習這些「常識」。我們必須能夠用電腦可以理解的方式，把知識呈現出來，包括一般性的常識，和特殊領域的專業知識。

由於在分析過程中，一般是依循斷詞、語法分析、語意分析等程序進行。但往往在做前一步驟時，就需要後面尚未執行之步驟所產生的訊息。例如在斷詞時，常常也需要使用句法及語意的訊息來協助判斷。因此在機器翻譯的過程中，若採用線性流水式的處理程序 (Pipelined Architecture)，則前面的模組經常無法作出確定性的 (Deterministic) 判斷，而須盡量多保留候選者，讓後面的模組處理。因此，最終判斷的時機應盡量延後，待累積足夠資訊後，再決定要使用的譯法。這樣才不會在資訊尚未完整的時候，就把正確的譯法排除到考慮範圍之外。

- 不合設定的語法

另外，雖然所有的語言都有語法，但一般我們所謂的語法，其實是一些語言學家，針對目前擁有的語料，所歸納出的一些規則。這些規則不見得完整，往往也有許多例外。再加上語言是一直在變遷的，因此我們無法要求語言的使用者，每字每句都合乎這些人訂定的文法，自然也難以避免這些狀況發生在我們所要處理的翻譯稿件中。這些不合設定語法的例子包括不明的字彙，如拼錯的字或新產生的專有名詞，和舊有字彙的新用法。例如「Please xerox a copy for me.」這樣的句子，即將影印機大廠 Xerox 的公司名稱當作動詞「複印」來使用。

這些狀況有些來自於單純的疏失，例如錯字、漏字、贅字、轉檔或傳輸時產生的亂碼，或是不慎混入的標籤 (tag)，也有些是已經獲得接受的新字彙和新語法。理想的機器翻譯系統，必須能夠適當地處理這些不合設定語法的問題。

除了字彙以外，在語句的層次也有可能出現不合文法的情形。例如「Which one?」之類的短句，在句法層次違反了傳統的英文文法，因為句中沒有動詞，不合乎許多文法課本對句子的定義。而「My car drinks gasoline like water.」這樣的句子，也違反了一般認為動詞「drink」的主詞必須是生物的設定。

(四) 解決方法

欲解決上述的歧義或語法不合設定問題，在在需要大量且瑣碎的知識。這些大量知識的呈現、管理、整合以及獲取 [Su 96]，將是建立機器翻譯系統時的最大挑戰。我們不但要將這些包含在語言學之內 (intra-linguistic)、跨語言學的 (inter-linguistic)，以及超乎語言學之外 (extra-linguistic) 的知識抽取、表達出來，用以解決上述的語法錯誤和歧義問題，還要維護這個龐大的知識庫。

此外，由上文可知，光是依靠專業領域的字典，仍然無法解決各領域的特殊問題。我們真正需要的，是各相關領域的專業知識。因此，我們要建立的知識庫必須包羅萬象，涵蓋各領域、各層面的知識。這些知識不但範圍廣大，而且雜亂瑣碎，要將它建立完善，本身就是一項艱鉅的工作。事實上，知識的取得是機器翻譯系統開發上最大的瓶頸。也因此，若要解決機器翻譯的問題，一定要有成本合宜且全面性的知識獲取方式，並兼顧多人合建系統時的一致性 (Consistency) 問題。

通常知識的獲取方式，和我們表現知識的方式有很大的關聯。表現知識的方式可以有不同的形式。例如一般的英文常識告訴我們，冠詞後面不會出現動詞。要表現這項知識，我們可以使用條列式的規則：「若某字是冠詞，則下一個字不可能是動詞」，也可以使用機率式的描述：「若某字是冠詞，則下一個字是動詞的機率為零」。這兩種不同的知識表達方式，會衍生出以下兩種不同的機器翻譯策略。當然除此之外，常用的還有儲存大量例句的例句式 (Example-Based) 系統，將不在此詳述。有興趣的讀者，可自行查閱 [Nagao 84] 等相關文獻。

- 規則庫方式

規則庫系統係由事先以人力建立好的大量規則所構成。進行翻譯的時候，電腦即依據這些規則，進行是與否的二擇判斷，以決定分析、轉換和生成步驟中，最後被選定的答案。這種作法也是早期大多數機器翻譯系統所採行的作法 [Hutchins 86]。

規則庫方式的優點在於貼近人類的直覺，因此容易了解，而且可以直接承襲現有的語言學知識和理論，充分運用前人研究的結果。相較於下文中提及的參數化方式，規則庫方式耗用的電腦硬體資源也比較少。但是相對的，規則庫方式也有它的缺點。規則庫系統是一連串是與否的二擇，但是自然語言中卻處處可以見到違反規則的例外。因此，當遇到複雜且較無規律的狀況時，規則庫方式往往就需要引用大量繁瑣的規則來處理。但規則的總數越多，維護起來就越困難。而且只要出現少部分無法精確區隔的例外情況，就會大幅降低整體的效能。例如若每個規則在進行判斷時的正確率可達 90%，則經過 20 次判斷之後，錯誤逐漸累積，其正確率就有可能銳減為 12% (0.9 的 20 次方)。因此規則庫方式一般說來僅適用於較為常規的狀況。

此外，規則庫式翻譯系統的建立和維護完全須仰賴人力，這也是一項很大的缺點。首先，在現代社會中，

大量人力代表昂貴的金錢，而且人的能力有其侷限，例如一般人在腦中能同時處理的事項，通常只有 5 到 9 項。因此在作修正時，往往無法同時考慮到規則庫中所有的規則，和是否適用於所有的語料。可是，若要提升全系統的效能，就必須對系統作整體的考量，否則就很可能會產生所謂的「翹翹板效應」(即某個範圍內的效能提升，反而使另一個範圍內的效能下降)，而無助於提升整個翻譯系統的效能。

上述這些缺點，使得規則庫翻譯系統的建立、維護和擴充十分不便。當系統的複雜度達到一定的水準後，翻譯品質往往就很難再行提升。這是因為規則庫方式的複雜度，在增加到某個程度後，就很可能會超乎人力所能維護的範圍之外。所以其效能常常在達到 70%至 80%的正確率後即停滯不前，很難更上一層樓。這些難題主要是來自於自然語言的特性，以及規則庫方式本身的缺陷。所以要突破這個瓶頸，我們可能得換個方式下手。

- 參數化方式

前文已提到，語言現象也可以用機率式的描述方式來表示。例如要表示冠詞不會接在動詞前面這個現象，我們也可以採用「冠詞的下一個字是動詞的機率為零」這個說法。若以數學式表示，即為 $P(C_i = \text{Verb} \mid C_{i-1} = \text{Det}) = 0$，其中 C_i 代表第 i 個字被歸為何種辭類。至於實際的機率值，則是來自以電腦統計語料庫中各種相鄰詞類組合 (如冠詞與動詞相連) 出現次數的結果，如下列公式所示：

$$W_1 W_2 W_3 W_4 \cdots (Words)$$

$$c_1\ c_2\ c_3\ c_4 \cdots (Part\text{-}of\text{-}Speeches) \qquad \cdots \Rightarrow P(\text{verb} \mid \text{det}) = \frac{\#[\text{det verb}]}{\#[\text{det}]}$$

在累積足夠的機率參數之後，就可以建立起整個統計語言模型。然後藉由參數之間數值大小的比較，告訴電腦人類在各種不同條件下偏好的解釋和用法。

這種機率表示法的最大好處，就是可以將參數估測 (統計) 的工作交給電腦進行。而且用連續的機率分布，取代規則庫方式中是與否的二擇，為系統保留了更多彈性。而估測參數時，由於是將語料庫中的所有語言現象放在一起通盤考慮，因此可以避免上述的「翹翹板效應」，達到全域最佳化的效果。參數化系統由大量的參數所組成，因此參數的獲取需要大量的電腦運算，儲存參數也需要相當大的儲存空間，超過規則庫方式甚多，但是在硬體設備發展一日千里的今天，硬體上的限制已經逐漸不是問題了。

採用參數化的方式，主要是因為自然語言本身具有雜蕪繁瑣的特性，有些現象無法找出明確的規則作為區隔，或是需要大量的規則才能精確區隔。為了能夠處理複雜的自然語言，機器翻譯系統也必須擁有能夠與之匹敵的複雜度。不過為了駕馭這些繁複的知識，我們還必須找到簡單的管理方式。但這是規則庫系統不易做到的，因為規則庫系統必需由人直接建立、管理，其複雜度受限於人的能力。若要增加複雜度，就必須增加規則數，因而增加系統的複雜度，甚至最後可能超過人類頭腦的負荷能力。參數化系統則將複雜度直接交由電腦控制，在增加複雜度時，參數的數量會隨之增加，但整個估測及管理的程序，則完全由電腦自動進行，人只需要管理參數的控制機制 (即建立模型) 即可，而將複雜的直接管理工作交給電腦處理。

在參數化的作法中，是將翻譯一個句子，視為替給定之原語句找尋最可能之目標語配對。對每一個可能

的目標語句子，我們都會評估其機率值，如下式所示 [Su 95]：

$$P(T_i \mid S_i) = \sum_{I_i} P(T_i, I_i \mid S_i)$$
$$\cong \sum_{I_i} \{ \left[P(T_i \mid PT_t(i)) \times P(PT_t(i) \mid NF1_t(i)) \times P(NF1_t(i) \mid NF2_t(i)) \right] \cdots (1)$$
$$\times \left[P(NF2_t(i) \mid NF2_s(i)) \right] \quad \cdots (2)$$
$$\times \left[P(NF2_s(i) \mid NF1_s(i)) \times P(NF1_s(i) \mid PT_s(i)) \times P(PT_s(i) \mid S_i) \right] \} \cdots (3)$$

上方的公式為參數化機器翻譯系統的範例，其中 S_i 為原語言的句子，T_i 為目標語的句子 (譯句)，I_i 為原語言-目標語配對的中間形式 (Intermediate Forms)，PT 為語法樹 (下標 s 為原語言，t 為目標語)，NF1 為語法的正規化形式 (Syntactic Normal Form)，NF2 為語意的正規化形式 (Semantic Normal Form)，而 (1)、(2) 和 (3) 三個列式，則分別代表生成、轉換和分析不同階段中的機率。

參數化系統還有一項極大的優點，就是可藉由參數估測的方式，建立機器學習 (Machine Learning) 的機制 [Mitchell 97]，以方便我們建立、維護系統，和依據個人需求自訂系統 [Su 99]。因為一般來說，如果能特別針對某一個特定的領域來設計專屬的機器翻譯系統，將有助於品質的提升。例如加拿大的 TAUM-METEO 氣象預報系統 [Hutchins 86, 92]，其英法翻譯的正確率可達 90%以上，至今仍運行不輟。但是在以往規則庫的做法下，由於規則須以人力歸納，成本相當高昂，所以無法針對各細分的領域逐一量身訂做專用的系統。但若採用參數化的做法，就可以使用不同領域的語料庫，估測出各式各樣的參數集。爾後只要更換參數集，便可將系統切換至不同的領域，以配合不同使用者、不同用途的需求。而且每次翻譯作業完成後，還可將使用者的意見納入新的參數估測程序中，使系統越來越貼近使用者的需要 [Su 99]。以下我們將進一步說明如何建立機器學習的機制。

(A) 非教導式學習

一般來說，要讓電腦進行學習，最直接有效的方式，就是將語料庫標注後，讓電腦直接從中學習標注的訊息，也就是所謂的「教導式學習 (Supervised Learning)」。但因標注語料庫需要花費大量的專業人力，且不易維持其一致性，所以對我們來說，最理想的機器學習方式，莫過於「非教導式學習 (Unsupervised Learning)」[Mitchell 97]，即不須人力參與，讓電腦直接從不加標注的語料庫中學習。

不過要達到非教導式學習的理想相當困難。因為自然語言本身會有歧義現象，在沒有任何標注資訊的情況下，電腦很難判斷文句的真意。為了降低學習的困難度，我們可以使用雙語的語料庫 (即原語言與其目標語譯句並陳的語料庫)，間接加上制約，以降低其可能之歧義數目。由於雙語語料庫中並列的原語言和目標語譯句，其語意必須是一致的，也就是雙方在可能的歧義上，必須求取交集。如此即可減少可能的歧義，讓電腦了解到句子的正確意思。

以「This is a crane. / 這是一隻白鶴。」這個原語言/譯句配對為例，「crane」一字在英文中有「白鶴」和「起重機」兩個意思。若單看句子，在沒有標注的情況下，電腦很難判斷出這裡的「crane」要作何解釋。但若給了中文的對應句子，那麼很明顯此處的「crane」指的一定是白鶴 (即兩者的交集)，才能使中英文句子表達的意思一致，因為中文的「白鶴」一詞並無「起重機」的歧義。在不同的語言中，詞彙

的解釋分佈通常是不一樣的，所以雙語語料庫中的配對，可以形成一種制約，有助於大幅縮減歧義的數量及可能範圍。

(B) 不同的參數化作法

在建立原語句和譯句的對映關係時，可以使用的方式有純統計方式 (又分 word-based [Brown 93] 和 phrase-based [Och 04] 這兩類)，以及使用語言學分析為基礎的語法 [Yamada 01] 或語意樹 [Su 95, 99] 對映。純統計方式是目前 IBM 模型 [Brown 93] 所採用的做法，其特徵為不考慮句子的結構，純粹以單字或詞串 (phrase，此處的詞串可以為任意連續字，不見得具有語言學上的意義) 為單位進行比對。這種方式的缺失在於只考慮局部相關性 (Local Dependency，通常為 bigram 或 trigram)，往往無法顧及句中的長距離相關性 (Long-Distance Dependency，例如句中的 NP-Head 與 VP-Head 通常會有相關性)。若兩個文法上有密切相關的單字之間，夾雜了很多其他的修飾語，就會使它們彼此超出局部相關性的範圍，此模式即無法辨識這種相關性。近來的 phrase-based 方式 [Och 04]，已針對上述缺點，改以詞串為單位進行比對，這樣雖然可以解決詞串內單字的相關性問題，然而在相關字超出詞串的範圍時，還是會產生無法辨認長距離關聯性的缺失。

但若使用以語言學知識為基礎的做法，不僅可以顧及語句中的長距離關聯性，而且句子的分析和生成結果，還可使用在其他用途上 (如資訊抽取、問答系統等)。如下方圖 2 中所示 [Su 95]，將原語句和譯句分別進行語法及語意分析，各自產生其語法樹及語意樹，再對所產生的語法樹或語意樹之各節點進行配對映射。但由於句子有歧義的可能性，每個句子都有數種可能的語法樹或不同的語意解釋，因此我們可以依照前文中的例子所述，藉由兩者間的對映關係，以採取交集的方式，分別排除原語言語法樹和目標語語法樹的歧義，如圖 3 所示。

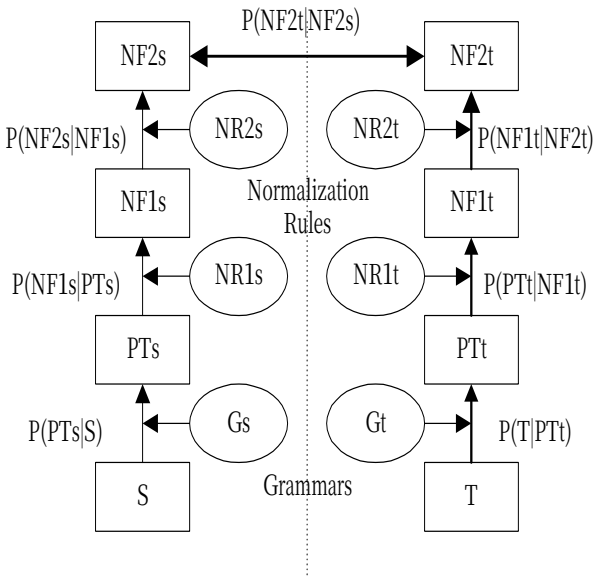


圖 2. 雙向式學習流程

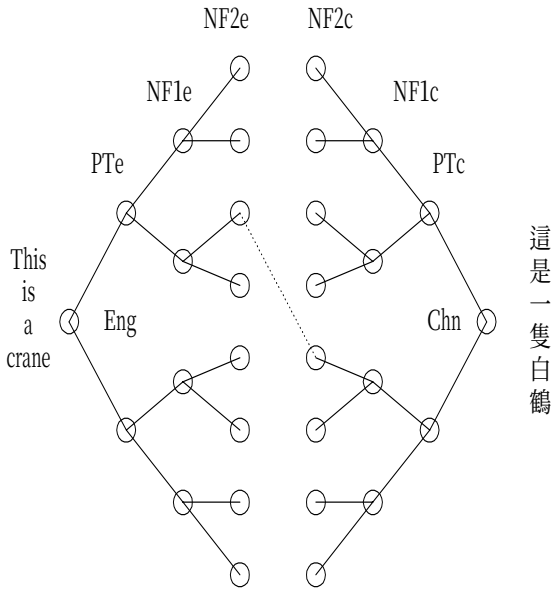


圖 3. 雙語配對句不同歧義間之映射

雖然在分析的過程中，由淺至深有許多不同的層次。理論上，原語言和目標語可在任一層次的結構上建立對映關係，如詞串到詞串、語法樹到詞串、語法樹到語法樹、語意樹到語意樹等。但事實上，採取不同的對映層次，會影響到對映的難易程度。如下方圖 4 所示，當在語法樹上做映射時，由於兩邊文法結

構不同，許多節點無法被對映到 (即圖中的白色節點)。然而當轉到語意層次做對映時，對映不到的節點 (白色部分) 就會減少很多，如圖 5 中的例子所示。在這個例子中，所有語意樹上的節點甚至全部都可以一一對映到。因此同樣的句子，採用較深層的語意層次進行雙向式學習，可以增加對映的效率。

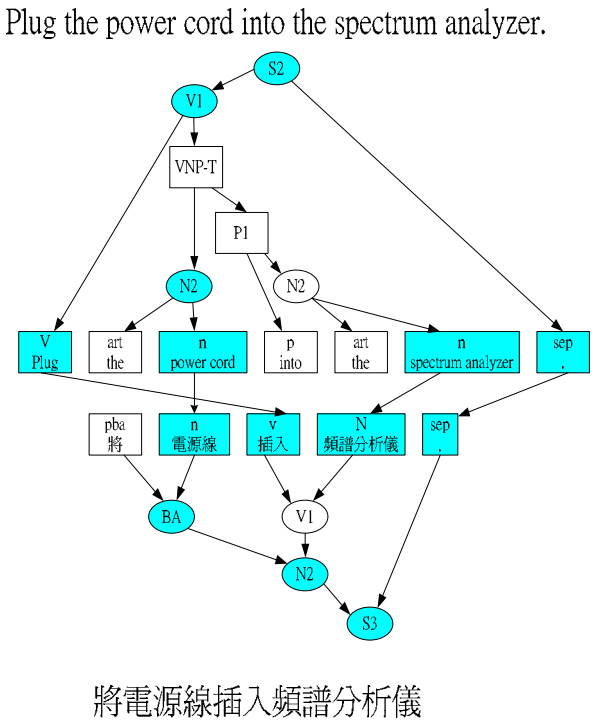


圖 4. 語法樹配對映射

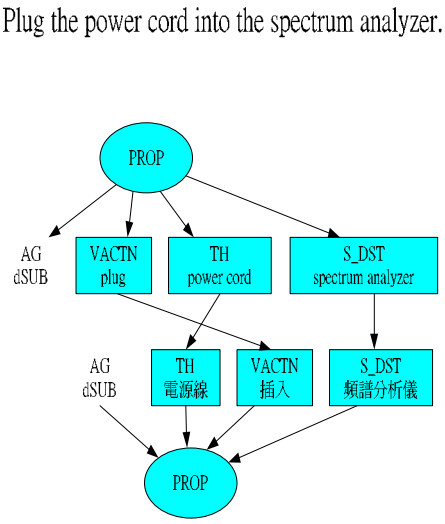


圖 5. 語意樹配對映射

上文論及若在語意層次進行映射，對映的效率較高。這主要是因為同樣的句子可以有不同的講法，如主動式、被動式等。所以配對中的兩個句子，可能會採用不同的講法，再加上不同的人寫出的原語言和目標語文法，其表達形式也可能有差異。因此如果直接在句法樹上作配對，對映效果往往很差。下表的實驗結果，也清楚呈現出這種趨勢。在 1531 句的句法樹配對 (PT) 中，只有 3.4%的句子擁有完全相符的語法剖析樹。但是如果先將這些語法樹轉成正規化的語意型式 (即下表中之 NF2)，甚至再做些局部的樹型調整 (如下表之 TC-TP，即 Target-Case-Topology-Tree)，則語意樹可完全對映的比例就可以提高到 50.3%。

	PT	NF1	NF2	NF3	TNF2LS	TC-TP
節點配對達成率	3.40% (52/1531)	11.23% (172/1531)	31.61% (484/1531)	32.72% (501/1531)	35.27% (540/1531)	50.29% (770/1531)

剩下無法完全對映的句子，經檢查後發現大部分其實語意已被譯者變更。如「Please check if the fuse is in the appropriate place.」，被譯為「請檢查是否已插入正確的保險絲」。嚴格來說這兩個句子所含的意思是不相等的。進行翻譯時，在多數情況下我們會希望譯句保有和原語句相同的語意，因此一般譯者會盡量維持語意相同。所以，先轉為正規化的語意形式，再行配對節點，可靠性會增加許多。

在將原語句和譯句配對後，所謂的自動學習過程，就是去尋找一組參數集 Λ_{MAX}，使其能讓所有原語句和譯句間之配對，有最大的「可能性」(likelihood value)。如下列公式所示 (其中 S 為所有的原語句，T 為所有譯句，I 則為所有分析過程中的中間型式)：

$$\Lambda_{MAX} = \arg\max_{\Lambda} P\left(T_1^N \mid S_1^N, \Lambda\right) = \arg\max_{\Lambda} \sum_{I_1^N} P\left(T_1^N, I_1^N \mid S_1^N, \Lambda\right)$$

這組參數即為參數化系統的「知識」，可以在翻譯的時候，用來決定哪一個目標語句最有可能是特定原語句的翻譯。由於參數化系統是以非決定性的方式來呈現語言現象，有別於規則庫系統的是/否二擇，因此保留了更多的彈性。這項特點在自然語言處理中十分重要，因為自然語言的歧義和語法不合設定問題，本身即具有非決定性的特質，因此較適合以非決定性的知識來解決。同時，參數化系統可藉由電腦的統計語言模型，自動從語料庫中學習有關語言的知識 (即機率參數)，大幅減低了建立和維護過程中需要的人力。隨著電腦化和網路的普及，語料庫的取得越來越方便，涵蓋的領域也越來越廣。參數化系統可以充分利用這項資源，作為其知識的來源，而無須太多的人力介入。基於上述的原因，近年在機器翻譯系統的研發領域中，參數化系統逐漸取代了過去的規則庫系統成為主流。

(五) 未來展望

上文中已提及，製作高品質的翻譯系統，需要的知識極為瑣碎而龐大。這些知識的獲取和管理，正是翻譯系統研發的重大瓶頸。從過去的經驗可知，這項工作的複雜度已超過人類所能直接控制的範圍，即使真的可行，其成本也不是大多數研發單位所能負擔的。

因此近年來機器翻譯系統的研發，已經逐漸由以前的規則庫方式轉為參數化方式。美國國家標準局 (NIST) 最近連續幾年，都針對中譯英的機器翻譯舉行評比。到目前為止在所有參賽系統中拔得頭籌的，都是參數統計式的系統 [NIST 05]，而且與其他類型的作法有不小的差距。由此可見，機器學習式的統計導向做法，已證明其優越性。目前機器翻譯研發的主流，已經逐漸從規則庫導向轉為參數統計方式。

這種典範轉移 (Paradigm Shift) 現象的產生，不只是因為大家認知到，機器翻譯系統的複雜度已超出人所能直接控制的範圍，部分原因也在於語料庫的發展規模。以往在建立語料庫時，是由人工從紙版資料打字鍵入，因此規模多半不夠大，對語言現象的涵蓋度也不夠高。所以主要是用來提供線索，供研究人員進一步將其概括化 (Generalize) 為通用的規則，以提高涵蓋範圍。但由於電子化的時代來臨，越來越多的文件是直接以電子檔產生，因此建立語料庫時僅須直接編輯電子檔，無須再經人工鍵入，建構成本大幅降低。加上網路逐漸普及，與日俱增的網頁也可以當作語料庫的來源。同時，共享語料庫的觀念也獲得普遍認同，許多大規模的語料庫，都可用很低廉的代價從美國 LDC (Linguistic Data Consortium，網址為 http://www.ldc.upenn.edu) 獲得。如此一來，語料庫對語言現象的涵蓋度已大幅增加，對以人工進行舉一反三的概括化規則歸納工作的需求，已經大幅降低。

上述這些龐大的語料庫，可以用來建立不同領域共用及各領域專屬的參數集。過去的翻譯系統，大多是以泛用的系統核心搭配不同領域的字典，企圖解決專門領域文件的翻譯問題，但是結果卻不如預期。原因已如上述，在解決歧義和語法不合設定的問題時，必須使用到該領域的領域知識 (Domain Knowledge)，無法單靠專門用語字典。有了大量的語料庫之後，我們可以從中挑選屬於各領域範疇的部

分，從中抽取相關之參數集，以解決領域知識的問題。

隨著硬體效能的大幅躍升，電腦的計算能力和記憶容量已經不再是機器翻譯系統研發的限制因素。同時語料庫的規模也與日俱增，如果由人來推導模型，讓機器在大量的雙語語料庫上，進行機器學習獲取大量參數，將可大幅降低知識獲取的複雜度，而這正是以往機器翻譯研發的瓶頸所在。展望未來，如果能在統計參數化模型上，融合語言學的知識，並能以更適當的方式從語料庫抽取相關知識，則在某些專業領域獲得高品質的翻譯，也是樂觀可期的。如此，則機器翻譯在實用化上的障礙，也終將獲得解決。

(六) 參考文獻

1. [Brown 93] Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer, "The Mathematics of Statistical Machine Translation: Parameter Estimation", Computational Linguistics, Vol. 19, No. 2, pages 263- 311, June, 1993.

2. [Hutchins 86]Hutchins, W. John, "Machine Translation: Past, Present, Future", Ellis Horwood Limited, 1986.

3. [Hutchins 92] Hutchins, W. John. and Harold. L. Somers, "An Introduction to Machine Translation", Academic Press, 1992.

4. [Mitchell 97] Mitchell, Tom. M., "Machine Learning", McGraw-Hill Companies, 1997.

5. [Nagao 84] Nagao, Makoto, "A framework of Mechanical Translation Between Japanese and English by Analogy Principle", Artificial and Human Intelligence, pages 173-180 , Amsterdam: North-Holland, 1984.

6. [NIST 05] MT Benchmark Tests, 1995, http://www.nist.gov/speech/tests/mt/index.htm, National Institute of Standards and Technology (NIST), USA.

7. [Och 04] Och, Franz Josef and Hermann Ney, "The Alignment Template Approach to Statistical Machine Translation", Computational Linguistics, Vol. 30, No. 4, pages 417- 449, December, 2004.

8. [Su 95] Su, Keh-Yih, Jing-Shin Chang, and Yu-Ling Una Hsu, "A Corpus-Based Statistics-Oriented Two-Way Design for Parameterized MT Systems: Rationale, Architecture and Training Issues", Proceedings of the Sixth International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-95), Vol. 2, pages 334-353, Leuven, Belgium, July 5-7, 1995.

9. [Su 96] Su, Keh-Yih, Tung-Hui Chiang, and Jing-Shin Chang, "An Overview of Corpus-Based

Statistics-Oriented (CBSO) Techniques for Natural Language Processing", Computational Linguistics & Chinese Language Processing, Vol. 1, No. 1, pages 101-157, August, 1996.

10. [Su 99] Su, Keh-Yih and Jing-Shin Chang, "A Customizable, Self-Learnable Parameterized MT Systems: The Next Generation", Proceedings of the Machine Translation Summit VII International Conference, pages 182-188, Singapore, September 13-17, 1999.

11, [Su 02] Su, Keh-Yih and Jing-Shin Chang, "Lecture for Statistical Natural Language Processing", Microsoft Research Asia, Beijing, China, August 17-18,2002. http://www.bdc.com.tw/kysu/c_2_microsoft_research_asia,_beijing.htm

12, [Yamada 01] Yamada, Kenji and Kevin Knight, "A Syntax-Based Statistical Translation Model", Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL), pages 523-530, Toulouse, France, July 6-11, 2002.