國立中興大學資訊科學與工程學系

碩士學位論文

中文轉客文語音合成系統中的文句分析模組之研究

Research on the Text Analysis Module in a Chinese to Hakka TTS System

指導教授：余明興 博士 Dr. Ming-Shing Yu

黃豐隆 博士 Dr. Feng-Long Huang

研究生：林昕緯 Hsin-Wei Lin

中華民國一百零三年十一月

本論文獲客家委員會104年度

客家研究優良博碩士論文獎助

國立中興大學 資訊科學與工程學系

碩士學位論文

題目：中文轉客文語音合成系統中的文句分析模組之研究

姓名：林昕緯 學號：7101056048

經 口 試 通 過 特 此 證 明

論文指導教授 余明興 黃聖隆

論文考試委員 劉燦雲
余明興
黃聖隆

中華民國 103 年 11 月 28 日

致謝

回顧兩年半的碩士生涯，昨日種種歷歷在目。雖然這段學習的路程走得有些顛簸、有些坎坷，但終究還是熬到了寫致謝辭的時刻。這一路走來，受到了太多太多人的幫助，不管是經濟上、學業上、生活上還是精神上，他們都是支持我繼續走到這裡的力量。

首先要感謝我的指導教授，余明興老師以及黃豐隆老師。沒有兩位老師的提攜之恩，也沒有今天的我。感謝余老師除了教導我做研究該有的嚴謹態度外，還不時的引導、啟發我的靈感，讓我找到研究的興趣與方向。

再來要特別感謝黃老師在研究期間的指導與大力協助，幫我四處收集客語語料，還時常督促我的研究進度與內容。也要感謝聯大的學弟妹們：佩俞、蓓瑩、易達、泓叡以及老同學俊明，謝謝你們辛苦的幫我標記語料，讓此論文得以順利完成。

也要感謝一起修課、做報告、吃飯的好同學：崇岳、誌暉，以及學弟們：星辰、時耀、世翔、允中、哲遠、順源，謝謝你們這段期間給予的幫忙與照顧。在此要特別感謝時耀在研究中的協助，讓系統更加完整。

最後，我要感謝我的家人以及女友，謝謝你們的支持與鼓勵，在我遇到低潮時給予我繼續奮鬥的力量，才能讓我順利完成學業。

中文摘要

本論文完成了一套中文轉客文語音合成系統，系統組成包括：文句分析模組、韻律訊息模組、語音合成模組。使用者輸入中文句子或文章，經過上述模組處理後，會輸出該句子的以下結果：1.客語斷詞及詞性標記結果、2.客語文句剖析結果、3.韻律階層預測結果、4.客語讀音求取結果、5.四縣腔的中文轉客文語音合成結果。

針對中文翻客文的斷詞處理，我們先蒐集客語句子語料，並設計一套工具，以半自動方式對客語句子做斷詞及詞性結果的標記。再利用標記結果訓練出國客語對應式的語言模型，最後應用我們提出的 Mix-Gram 分數算法於斷詞候選序列的選擇。經實驗結果顯示，在嚴重資料稀疏的情況下，此方法的正確率有 81.41%。

而客語讀音求取方面，我們採用照詞或字翻譯的方法，針對客語發音辭典，我們將每詞彙切分為單字，並抽取其 1.讀音、2.詞性、3.是否為詞尾等三個特徵，訓練出讀音資訊統計表。以客語詞彙發音辭典為優先,該統計資訊為輔的方式,設計出一個客語讀音求取的流程。經實驗顯示，此方法對客語讀音求取的正確率有 82.81%。

最後，我們實際合成出客語語音，並做平均主觀分數的測試。結果顯示，語音合成系統中的韻律訊息及文意正確性有明顯的改善。

關鍵詞：中文轉客語斷詞處理、文句分析、讀音求取

Abstract

This thesis presents a Chinese text to Hakka speech system. It consists of the following modules: text analysis module, prosodic analysis module, and speech synthesis module. The input data is Chinese text or article which will be processed by above modules. The following results are produced: 1. Hakka word segmentation and part of speech, 2. Hakka syntactic tree, 3. Prosodic phrase prediction, 4. information for Hakka pronunciation, and 5. speech result of Hakka Si-Xian(四縣).

With the methods for Chinese to Hakka words mapping and segmentation, we collected the Hakka sentence corpora. We designed a tool for semi-automatically mapping the word segments and tagging part of speech. Then we trained a language model from the mapping and tagging results. Finally, we apply the proposed mix-gram method for text analysis in this thesis. In final evaluation, the experimental result demonstrates about 81.41% precision rate, under the circumstance of data sparseness.

For the problem of Hakka pronunciation, we apply the word-based method, and translated each word to pronunciation by using a Hakka dictionary. We gather the following three factors: 1. Pronunciation, 2. Suffix, and 3. Part of speech, as our statistic data. Combined with Hakka pronunciation dictionary, we extract the Hakka pronunciation of text input by user. The experimental result shows 82.81% precision rate in extracting Hakka pronunciation.

Finally, we evaluated the synthesized Hakka speech by mean opinion score, MOS. The experimental result shows that our approach can significantly improve the performance in a Chinese text to Hakka speech system.

Keywords: Word Segmentation of Chinese to Hakka, Text Analysis, Grapheme to Phone Conversion

目錄

表目錄

圖目錄

第一章 緒論

現有客語語音合成系統相關的研究文獻，較著重於客語語音合成系統的建置方法，以及最後端的語音訊號處理。對於前端客語文句分析模組的研究，以及客語文字語料的處理，並沒有太大的進展。最主要的原因是，客語語料的採集非常困難。現有較能輕易取得的客語文字電子資料，如：客委會初級[14]、中高級的認證教材[12][13]、教育部編著的國小客語教材[24]…等，對於電腦自然語言處理來說，資料規模仍然屬極小量語料。若想要建置出更多的客語語料，都需要從客語書籍、文章中，透過人工輸入成電子檔的方式來取得，相當費工與耗時。但是，即便有了這些基本的客語文句語料也只是研究的第一步，若要應用自然語言處理技術做更進一步的分析與應用，後續仍有許多的處理工作，如：人工斷詞、詞性特徵的標記、語言模型的建置、文句剖析樹的標記…等。

客語斷詞研究方面，依現有的文獻顯示，目前中文轉客文的客語斷詞處理方法，因沒有足夠的客語語料來建置客語語言模型，其處理方式都是採用中文詞查找國客語對照詞典(Word-Based)來直接轉換的方法。此方法正確率低，且缺乏文句中語意的資訊。通常轉換出來的客語句子，都會失去客語應該有的「語氣」，變成「國語式客語」的感

覺。因此本論文提出一套針對中文轉客文語音合成系統中的文句分析模組之研究方法,期望能改善客語語音合成系統中,其文意的正確性。我們建置語料的方法及成果，也能提供後續客語相關論文，如：智慧型客語輸入法、客語語音辨識…等來使用。

1.1 研究動機與目的

根據行政院客家委員會在 2010 年至 2011 年的調查[11],台灣的客家族群占了 18.1%比例，約有 419.7 萬人，人口數量僅次於閩南族群。但客家族群會講母語的比例，卻遠不及閩南族群。而 2004 年客語使用狀況調查報告也明確的指出[10],阻礙任何一種文化傳承的最大禍首，是語言的失傳。因此要保存客家文化的首要工作，就是推廣客語的學習。語言的學習，包括了：聽、說、讀、寫。針對聽與讀，我們實驗室已發展了線上客語有聲辭典,系統提供使用者於線上查詢客語詞彙、國客語之例句及其發音標記，以及中文句子的文句轉客語語音之功能。

然而，對於一個語音合成系統而言，比後端語音處理更根本的問題，是前端文句分析處理時，能夠輸出正確的文意訊息及韻律訊息，才能將句子唸對、符合文意，合成出正確的語音。因此，本論將探討如何建置客語語料及客語語言模型，以及探討如何實現一個較理想的

中文轉客文之客語斷詞處理方法。期望除了提供目前的線上有聲客語辭典使用外，也能提供更多不同的系統應用，如：客語有聲電子書、客語新聞播報…等。

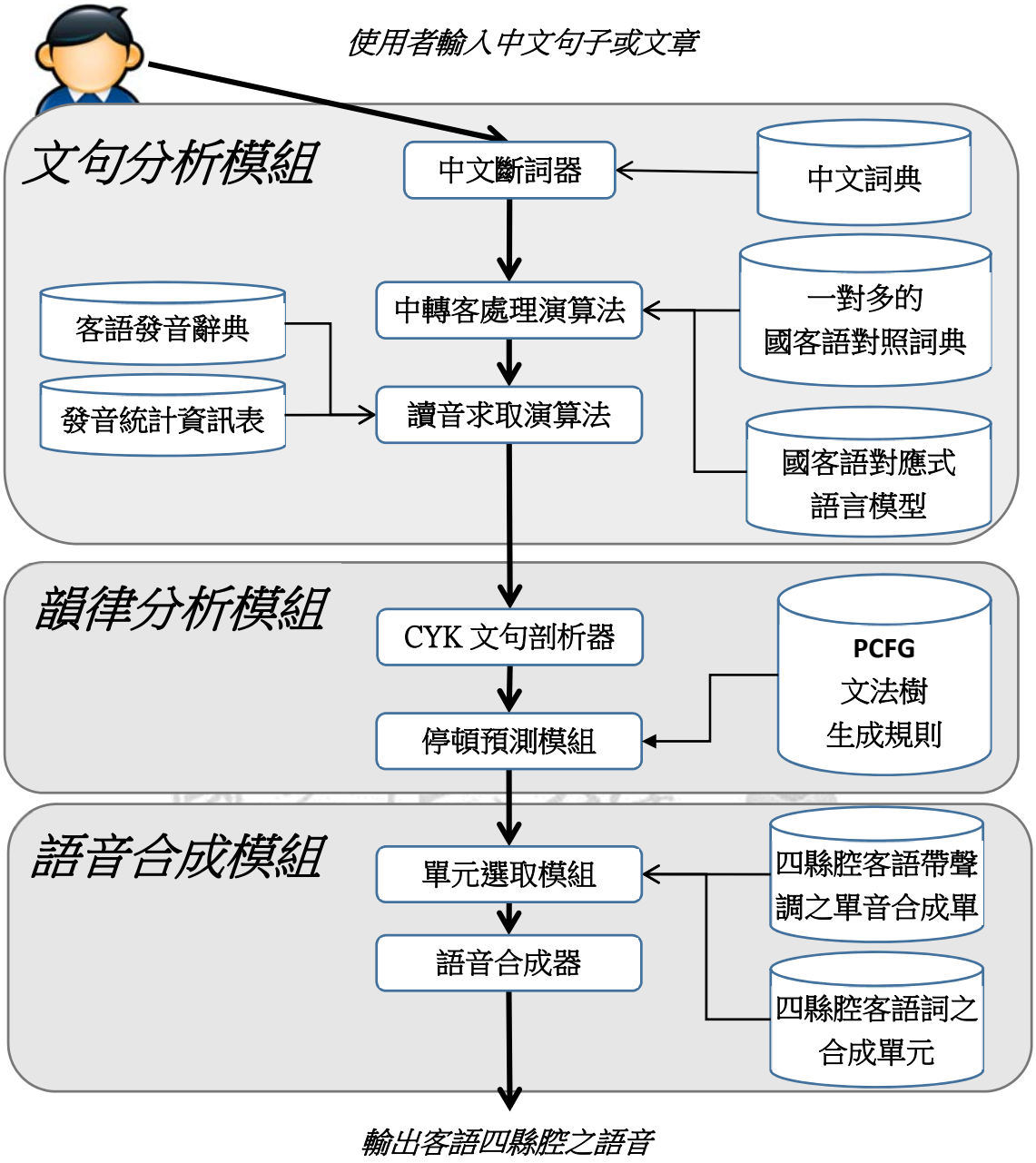
1.2 研究方法概述

本論文主要是針對客語文句分析模組做進一步的研究探討，要改善此模組最直接的影響是語料的大小。然而，目前能取得的客語語料，皆是一些尚未做斷詞處理的文章及句子，還未有現成的客語斷詞語料能使用。因此我們的研究項目，主要包括：

1. 客語斷詞語料的標記方法及客語語言模型的建置方法
2. 客語斷詞的處理方法
3. 客語讀音的求取方法

除了以上重點研究項目外，因為本論文也包含實作出整個中文轉客文語音合成系統，我們必須在句子的詞間加入適當的停頓訊息、讓語音更自然、更容易聽懂。因此在後端韻律訊息模組部分，我們也結合本實驗室在中文語音合成系統(Mandarin TTS System)中的韻律階層求取之研究成果，來求取客語句子中的韻律階層，得到該句子的無停頓(No break)、小停頓(Minor break)及中停頓(Major Break)之韻律訊息。會直接採用中文的方法，主要是因為客語句子與中文句子的文法結構

幾乎相同，且客語句子目前尚未有足夠的韻律階層標記之語料，能做停頓預測模型的訓練。我們的客語用字，是採用教育部的「臺灣客家語推薦用字[23]」。而拼音方案是採用教育部所制定的「客家語拼音方案[22]」。在客語斷詞語料方面，我們使用客委會所發行的「四縣腔初級[12]、中級暨中高級客語認證教材[13][14]」中的例句，來標記客語斷詞資訊，以及訓練客語語言模型。圖一為系統架構圖，其中「中文斷詞器」與「中轉客處理演算法」，之後合併稱為「客語斷詞處理」。



圖一：中文轉客語語音合成系統之系統架構

1.2.1 客語斷詞的標記及語言模型的建置

我們認為，客語詞的判定是一件嚴謹的事情。理論上我們必須遵照詞的定義[1]來標記，但有少數的情況下，我們仍會將詞組標記成一個

[1] 詞(Word)，是指最小、有完整明確意義且可以自由使用的中文語言單位。

客語詞，如：滑溜溜，在中文斷詞被斷為兩個詞：滑/溜溜。但我們在標記時視它為一個詞：滑溜溜。而對於非客語語言專家的標記人員來說，其最有效率的方法，是透過具有平行資訊[2]的語料，先將中文語料輸入至中文的文句處理系統，取得中文的斷詞、詞性標記的特徵後，再對其對應的客語文章，以人工方式去判斷客語詞的邊界與詞性的標記。這個方法普遍被使用於同類型[3]的平行語料標工作記上，如蔡依玲的碩士論文[31]也是用此方法。這是因為中文斷詞的技術已經相當成熟，而客語與中文的文法結構也相近，實際上中文的斷詞、詞性特徵，幾乎都能直接對應於客語詞。

為了快速建置客語斷詞語料，我們開發了一個便於操作的客語斷詞答案標記工具，結合一個正確率有 96.69%的中文斷詞模組[32]，自動針對中文句子做斷詞處理，標記者再依據該中文斷詞之結果，使用滑鼠在客語句子上做反白之操作，即可標記出斷詞結果(詳見第四章)。針對標記者工作時，可能會遇到不知如何標記的情況，為此我們也制定出五個簡單的標記原則，讓標記者能快速、準確的標記出客語斷詞及詞性標記結果。

最後，我們透過該標記結果，取部分做為訓練國客語對應式的客語語言模型的語料，部分做為測試之語料。並使用了 Additive

[2] 具有中文文句和客文文句 1 對 1 對應的平行資訊的語料。
[3] 客語與中文都屬於漢語，文法結構幾乎相同，僅有少數俚語、特殊的客語構詞不同。

Smoothing、Katz Smoothing 及 Enhanced Katz Smoothing[16]等語言模型平滑方法,找出較理想的語言模型平滑方案。最後根據實驗結果觀察，我們提出了 Bi-Gram 與 Uni-Gram 混合計算的 Mix-Gram 候選序列分數算法。

1.2.2 客語斷詞的處理方法

我們提出了一個中文轉客文斷詞處理的方法，透過兩階段方式，將使用者輸入的中文文句，先以中文斷詞模組得到第一階段的斷詞及詞性標記結果。再來，將第一階段的結果，以國客語對照辭典找出所有可能被轉換的客語詞，並建立出所有可能的客語斷詞候選序列。最後，再以客語 Bi-gram、Uni-gram 之國客語對應式語言模型，配合我們提出的 Mix-Gram 分數算法來選擇出分數最高的客語斷詞候選序列，並輸出為客語斷詞結果。

1.2.3 客語讀音的求取方法

我們參考連又箴的碩士論文[25],以人工方式標記一部客語詞發音詞典，以及建置一張客語單字發音的對照表。針對客語發音辭典，我們再將每個詞分割為單字,抽取其 1.讀音、2.出現位置、3.詞性等特徵，建立統計資訊。並也以同樣的特徵與方法，另外針對多音字建立統計資訊。透過上述的統計資訊，以客語發音辭典為主、統計資訊為輔的

方式，建立四段式的求取流程，來對客語讀音做預測。

1.3 文獻探討

1.3.1 客語語料建置的相關研究

目前有針對客語斷詞做人工標記的論文不多，其中之一為交大電信工程所蔡依玲的碩士論文[31]。他們的語料採用為龔萬灶老師所撰寫的「阿啾箭个故鄉」一書內的客語文章。將其中包含 42 篇文章，委由余秀敏老師將此客語文句語料翻譯成相對應之中文平行語料。並將翻譯後的中文文章以中文斷詞系統取得中文斷詞及詞性標記的特徵後，再以人工標記上客語斷詞及詞性等資訊。最後得到的語料，包含標點符號共有 63158 個字、36450 個詞。但該語料並沒有用來建置客語語言模型，且許多客語用字皆與目前教育部建議用字[23]不同。

另外，有屏東教育大學的「學術研究基礎建置暨客家文化研究計畫[33]」。他們歷時至少三年的時間，在收集、建置客語語料及詞頻庫。這項工作有助於客語文句處理的發展，如：客語斷詞系統、客語文句分析系統、客語文句剖析系統、客語語音合成系統、智慧型的客語輸入法…等；上述工作都非常需要足夠的客語語料來支持其研究與發展。

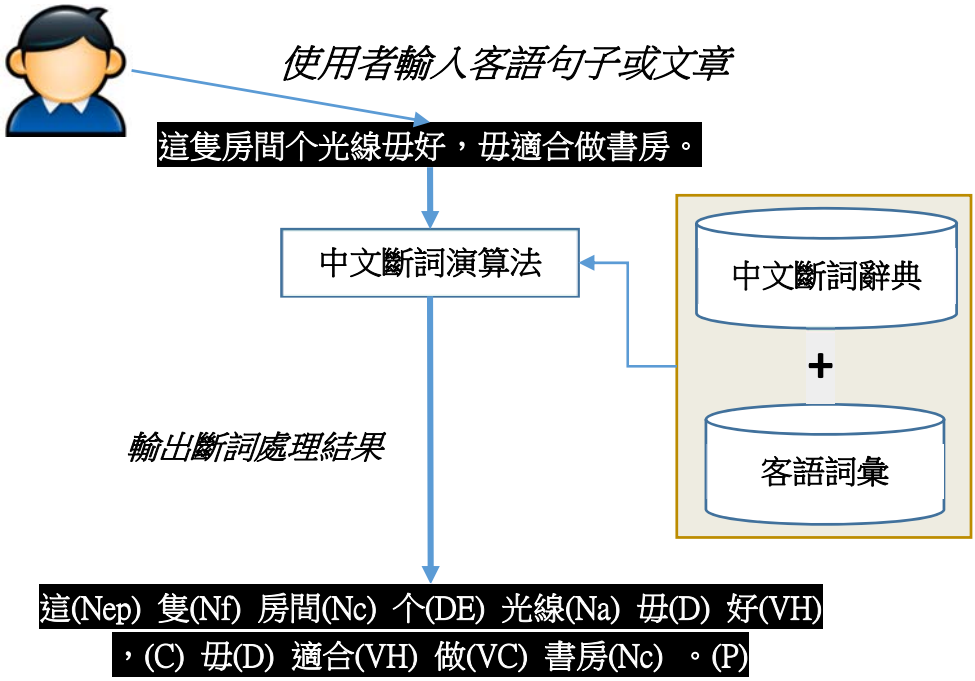
1.3.2 客語斷詞處理的相關研究

因客語語料非常少，且標記、建置之工作費時又費工的緣故，目前針對中文轉客文的相關研究非常少。客語斷詞的處理，基本上可以(一)輸入為客語、(二)輸入為中文。第一類是直接針對客語文句或文章做斷詞，第二類是針對中文文句轉換成客語斷詞之結果。

而目前國內外仍沒有任何論文，針對客語斷詞做深入之研究，其中評估客語斷詞效能的論文，僅有蔡依玲的碩士論文[31]。顯見目前客語斷詞的研究，不管是語料的建置，還是斷詞的方法，仍有非常多待探討與解決的議題。

(一)輸入為客語

這一類的系統，適合具備客語輸入能力及熟悉客語的使用者，對於一般不熟悉客語的使用者而言，較不方便。這類系統常見的做法，是直接使用中文斷詞系統，對客文做斷詞。當然，這樣會有一些客語造字或客語用詞無法辨別的問題，針對這部份，是使用國客語對照辭典，來解決客語未知詞(Out of Vocabulary, OOV)問題。圖二為輸入是客語文句的客語斷詞處理流程之示意圖。



圖二：輸入為客語的客語斷詞處理方法示意圖

如蔡依玲的論文[31]，他們透過 Conditional Random Field 方法實作中文斷詞系統，並於系統中加入國客語對照之外部斷詞辭典。最後結合少量的客語構詞規則，實做出客語斷詞模組。其實驗結果顯示，客語斷詞的 F-Measure 為 82.87%，客語詞性標記的 F-Measure 為 77.14%。

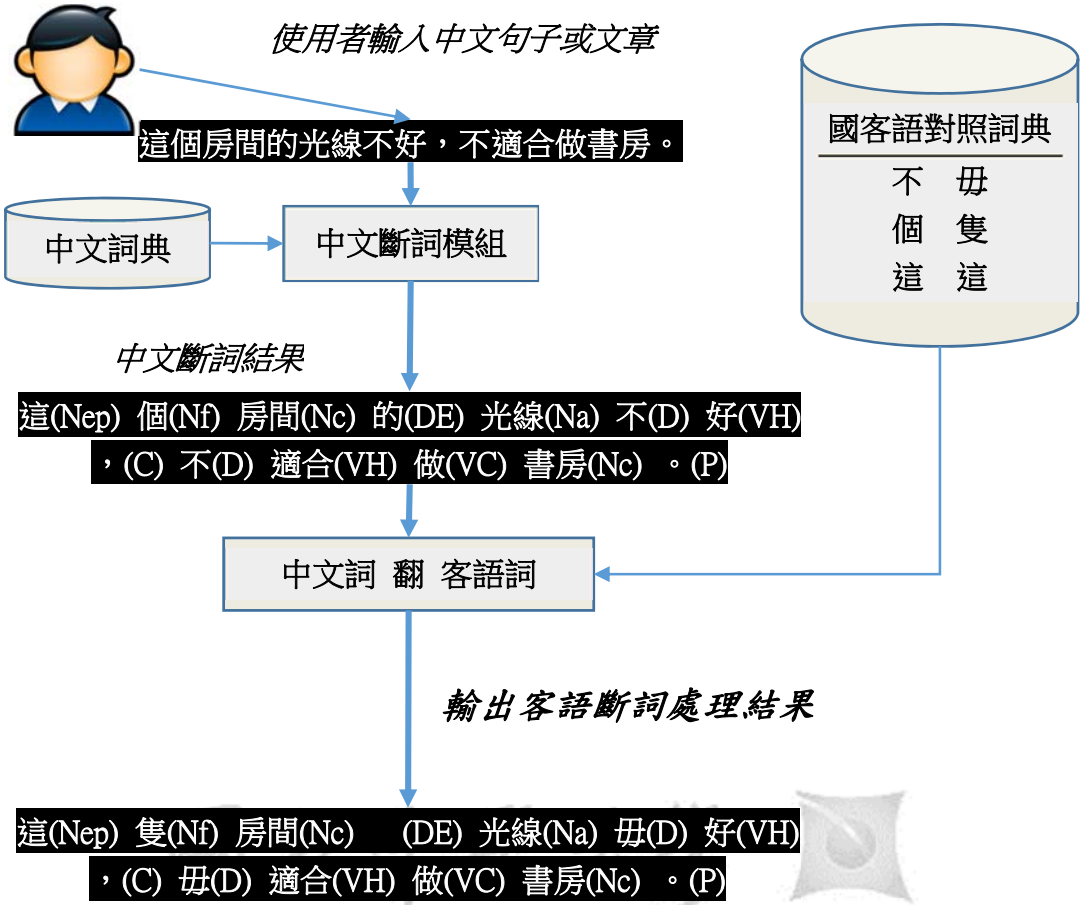
另外必須一提的是，我們論文所提出的客語斷詞處理方法，輸入為中文句子，輸出為客語斷詞及詞性標記結果。不同於輸入為客語句子的系統，因為要考慮到中文及客語轉換的方法，因此正確率會較低。但實際使用上，對於客語初學者將會更利於操作與學習。因為不需要另外學習客語拼音及客語輸入法的使用。

(二)輸入為中文

這一類的系統，使用者不需使用客語輸入法，也不需熟悉客語，很適合客語初學者使用。這類系統常見的做法，是使用中文斷詞系統，先將輸入的中文文句斷詞，找出詞與詞性後，再將詞透過國客語的平行對照辭典，翻譯成客語詞。如本實驗室的線上客語語音合成系統，吳俊毅[15]及羅丞邑[34]的碩士論文的斷詞方法皆相同，都是使用江昶毅的碩士論文[9]所提出的中文斷詞系統，將中文文句斷詞後，再透過國客語對照辭典，將中文詞翻譯成客語詞。經測試後，其不含詞性標記的客語斷詞效能的 F 分數分別為 69.82%及 66.72%。

另一種是僅透過國客語對照辭典，將中文文句直翻成客語。如李雪貞的碩士論文[17]，他們建置出一套國客語對照辭典，將輸入的中文文句字串切割成 1 到 4 字詞，並查找對照辭典、翻譯成客語詞。而他們沒有針對中文翻客語詞做效能評估，因此無法得知效果如何。

圖三為輸入是客語文句的客語斷詞處理流程之示意圖，該流程是傳統的處理方法，並未使用客語語言模型。



圖三：輸入為中文的客語斷詞處理方法示意圖

1.3.3 讀音求取的相關研究

目前的客語讀音求取尚未有相關的研究，本實驗室連又箴的碩士論文[25]，針對台語讀音求取的方法已有不錯的解決方案。台語讀音與客語讀音的求取方法雷同。這篇論文使用以詞或字找音(Word-Based)的方法為基礎，並對發音辭典中每個詞切分為單字，抽取每個單字的「是否為詞尾」、「詞性」、「讀音」等特徵資訊做統計，來輔助求取台語的讀音。最後，他們建立了一個線上台語讀音人工標記的平台，期

望透過人工為主，方法為輔的方式，快速建置出大量的台語讀音資料庫。該論文所提出的方法，對台語讀音求取的正確率有 80.6%。

1.4 論文架構

本論文共分為八章，章節概述如下：

第一章 緒論：完整的介紹本論文的研究動機與目的，以及做簡單的研究方法概述。讓讀者先概略性的了解，本論文的內容及架構。

第二章 客語四縣腔介紹：說明我們所實作的中文轉客文語音合成系統中，所採用的客家語拼音方案，以及變調規則。

第三章 語料及準備工具：介紹本論文研究過程中所用到的工具及語料。

第四章 國客語對應式語音模型建置方法：完整的說明我們如何收集客語語料，以及如何處理客語語料。並且介紹語言模型的相關理論基礎，以及國客語對應式的語言模型建置方法。

第五章 客語斷詞處理方法：完整的說明什麼是客語斷詞，以及我們所定義出的客語斷詞處理的分類。並且比較傳統方法與我們所提出的方法之差異，以及說明我們客語斷詞處理的實作方法及實驗結果。

第六章 客語讀音標記及求取方法：完整說明我們針對客語發音語料的處理，以及擴充的方法。並且具體敘述我們求取客語讀音的方法，以及實驗結果。

第七章 中文轉客文語音合成系統實作：完整的說明本論文所實作的語音合成系統，其環境、系統架構與實作方法，以及聽測實驗結果與分析。

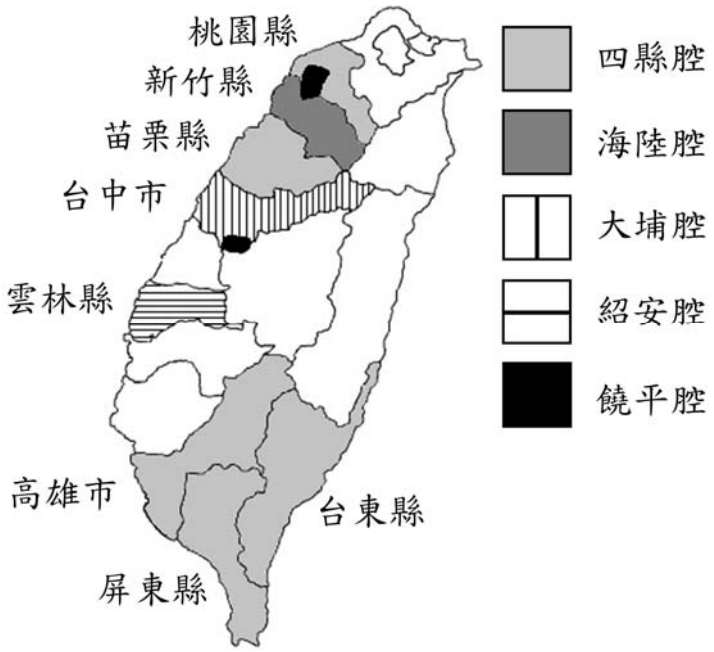
第八章 結論與未來改進方向：對本論文所提出的文句分析模組的研究方法做結論，以及提出未來可持續研究或改進的方向。

第二章 客語四縣腔介紹

本章將介紹在台灣使用客語的人口分佈，以及我們所採用的客語拼音方案，及其變調規則。

2.1 台灣客語分佈

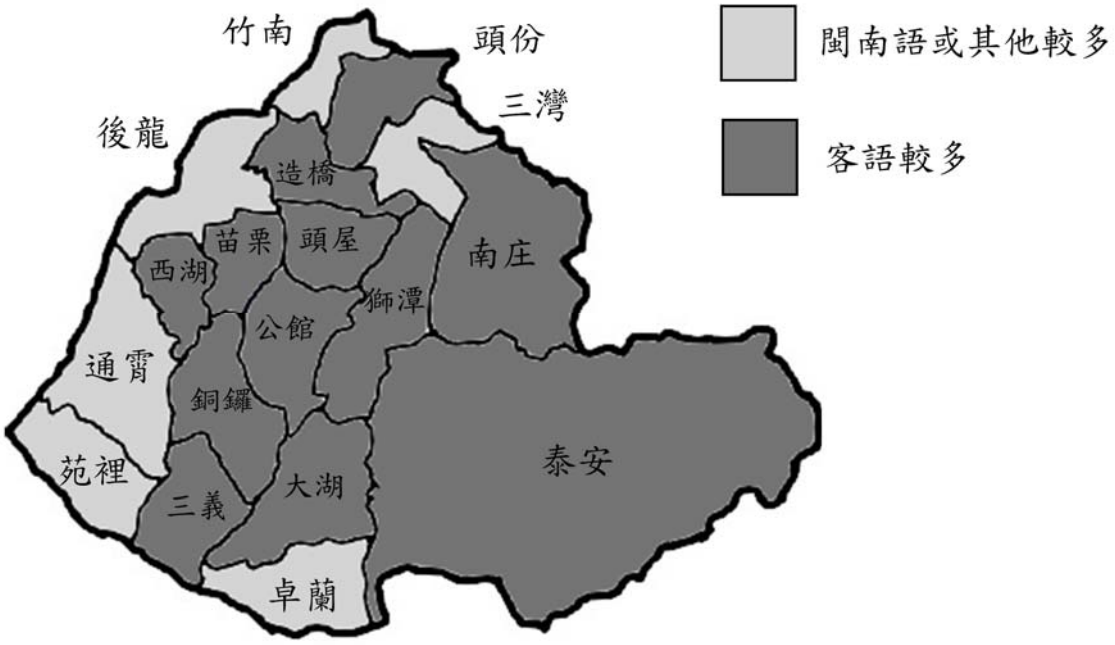
客語為台灣客家族群的母語，其主要源於中國大陸粵東，因地區的不同而形成了許多種客語腔調。在台灣所使用的腔調主要有五種，分別為：1.四縣腔、2.海陸腔、3.大埔腔、4.饒平腔、5.詔安腔，其中又以四縣腔最為盛行。因本論文重點在於客語語料處理及客語斷詞處理的一系列之方法，因此先以四縣腔為主來發展研究方法。其餘的客語腔調，甚至是各種漢語系的方言，未來皆可參考本論文之方法來建置語料與發展相關系統。我們可由圖四：客語腔調使用分佈圖觀察出，四縣腔為目前使用最廣泛之客語腔調。



圖四：台灣客語腔調使用分佈

四縣客語指的是大陸廣東省之蕉嶺、平遠、五華、興寧四個地區所通行的客語腔調。在台灣使用四縣腔的客家人居多，因此即便是熟悉不同腔調的客家人，大部分也都還能用四縣腔溝通、或聽得懂四縣腔。

在台灣客家人分佈最密集的縣市為苗栗縣，是客家人的大本營，使用的客語大部分為四縣腔。苗栗縣除了靠海的苑裡鎮、通霄鎮、竹南鎮、後龍鎮與山區的三灣鄉、卓蘭鎮等地的方言[35]，使用閩南語或原住民語居多之外，其它鄉鎮大部分都使用客家話，如圖五所示。另外在北部的桃園縣中壢市、平鎮市、龍潭鄉等地區，也有非常多使用四縣腔的客家人。而南部高雄的美濃鎮，屏東縣的長治、新埤、萬巒、竹田、佳冬、高樹等地所使用的客語屬於南四縣腔。



圖五：苗栗縣使用客語之分佈

2.2 客語拼音方案

本論文所使用之客家語拼音方案，為教育部所公告的台灣客家語拼音方案[22]， 最近一次的更新為 2012 年 11 月 29 日的修正公告。

客家話的音節結構和其他漢語方言(如閩南語)一樣，可以分為兩大部分：1.聲母和韻母及 2.聲調，聲母是指音節的第一個輔音，而韻母又可分為韻頭、韻腹與韻尾。在音節結構中，只有聲調和韻腹是不可或缺的要素，其它則可有可無。表一為客家語拼音方案中的聲母符號表、表二為客家語拼音方案中的韻母符號表。

表一：客語聲母符號表

客家語拼音	b	p	m	f	v	bb [註 1]
國際音標	[p]	[p^h]	[m]	[f]	[v]	[b]
注音符號	ㄅ	ㄆ	ㄇ	ㄈ	万	ㆠ
客家語拼音	d	t	n	l	r [註 2]	g
國際音標	[t]	[t^h]	[n]	[l]	[j]	[k]
注音符號	ㄉ	ㄊ	ㄋ	ㄌ		ㄍ
客家語拼音	k	ng	h	j [註 3]	q [註 3]	x [註 3]
國際音標	[k^h]	[ŋ]	[h]	[tɕ]	[$tɕ^h$]	[ɕ]
注音符號	ㄎ	ㆭ	ㄏ	ㄐ	ㄑ	ㄒ
客家語拼音	z	c	s	zh [註 4]	ch [註 4]	sh [註 4]
國際音標	[ts]	[ts^h]	[s]	[tʃ]	[$tʃ^h$]	[ʃ]
注音符號	ㄗ	ㄘ	ㄙ	*ㄓ	*ㄔ	*ㄕ
客家語拼音	rh [註 4]					
國際音標	[Z]					
注音符號	ㆮ					

註解：

1. bb 可用於雲林詔安腔、南投國姓鄉及部分南部客家地區。
2. r 為摩擦音，僅用於部分南四縣腔。
3. j、q、x 可用於四縣腔及南四縣腔。
4. zh、ch、sh、rh (ㄓ、ㄔ、ㄕ、ㆮ)用於海陸、饒平、詔安等三腔。
   zh、ch、sh、rh (ㄐ、ㄑ、ㄒ、ㆮ)用於大埔腔。

表二：客語韻母符號表(單母音)

客家語拼音	ii	i [註 1]	e	ee [註 2]	a	o
國際音標	[ï]	[i]	[e]	[ɛ]	[a]	[o]
注音符號	�†††	ㄧ	ㄝ		ㄚ	ㄛ
客家語拼音	oo [註 2]	u [註 1]	er [註 3]	m [註 4]	n [註 4]	ng [註 4]
國際音標	[ɔ]	[u]	[ə]	[m]	[n]	[ŋ]
注音符號		ㄨ	ㄜ	ㄇ	ㄋ	ㄫ
客家語拼音	b [註 4]	d [註 4]	g [註 4]	nn [註 5]	m [註 6]	n [註 6]
國際音標	[p]	[t]	[k]	[~]	[m̩]	[n̩]
注音符號	ㄅ	ㄉ	ㄍ		ㄇ	ㄋ
客家語拼音	ng [註 6]					
國際音標	[ŋ]					
注音符號	ㄫ					

註解：

1. i、u 可用於韻頭、韻腹及韻尾。
2. ee、oo 僅用於詔安腔。
3. er 用於部分海陸、饒平。
4. -m, -n, -ng 用於陽聲韻尾(鼻音韻尾)；-b, -d, -g 用於入聲韻尾(塞音韻尾)。
5. nn 一般僅用於詔安腔，但其他腔亦偶可見，如：歪 uainn+(大埔)。
6. 輔音 m, n, ng 可視為韻腹，自成音節。如：(四縣)魚 ng∨。

2.3 客語聲調與變調規則

國語與客語都是聲調語言，同樣的拼音配上不同聲調會產生不同的意義，國語使用了五種聲調，而客語四縣腔使用了六種聲調，分別為：陰平、陽平、上聲、去聲、陰入、陽入。下表三為客語四縣腔聲調符號表。

表三：客語四縣腔聲調表

調類	陰平	陽平	上聲	去聲	陰入	陽入
調值	24	11	31	55	21	5
調型	fu´	fuˇ	fu`	fu	fug`	fug
例字	夫	扶	虎	富	福	服
近似國語聲調	2 聲ノ	3 聲∨	4 聲丶	1 聲		
音檔調號	2	3	4	1	4 [註 1]	1 [註 2]

註解：

1. 原音檔調號為 5，本論文依照其調型「丶」，皆改為「4」來表示。
2. 原音檔調號為 2，本論文依照調型，皆改為「1」來表示。

客語有連音變調的問題，我們系統在求出客語字的讀音後，必須做適當的連音變調，才能念出正確的讀音。而四縣腔可以歸納出三種連音變調的規則，如表四所示。

表四：客語四縣腔連音變調規則表

規則 1：由兩個陰平字構成的字彙，讀時前字變調讀陽平 陰平（˧）＋陰平（˧）→ 陽平（˅）＋陰平（˧）			
範例	詞彙	變調前之拼音	變調後之拼音
	新衫	xin˝ sam˝	xin˅ sam˝
	買新衫	mai˝ xin˝ sam˝	mai˅ xin˅ sam˝
規則 2：陰平與去聲構成的詞彙，讀時前字變調讀陽平 陰平（˧）＋去聲 → 陽平（˅）＋去聲			
範例	詞彙	變調前之拼音	變調後之拼音
	針線	ziim˝ xien	ziim˅ xien
	拿針線	na˝ ziim˝ xien	na˅ ziim˅ xien
規則 3：陰平與陽入字構成的詞彙，讀時前字變調讀陽平 陰平（˧）＋陽入 → 陽平（˅）＋陽入			
範例	詞彙	變調前之拼音	變調後之拼音
	音樂	im˝ ngog	im˅ ngog
	聽音樂	tang˝ im˝ ngog	tang˅ im˅ ngog

第三章 語料及準備工具

本論文所實做的中文轉客文語音合成系統有三大模組，分別為：1.文句分析模組、2.韻律分析模組、3.語音合成模組，其中文句分析模組的客語斷詞處理方法有使用到中文斷詞器，而韻律分析模組有使用到 CYK 中文文句剖析器及韻律階層求取器。詳細的關係圖，如圖一的系統架構圖所示。本章將介紹系統中所使用的工具及語料庫。

3.1 準備工具

3.1.1 中文斷詞器

本論文採用的中文斷詞器，是賴亦傑於 2011 碩士論文所提出的中文斷詞方法[32]。此方法係應用多詞(nWord)及多詞性(nPOS)的語言模型，實作兩階段式的斷詞方法。

第一階段是找出中文句子的斷詞邊界，其方法是使用 1Word、2Word 之語言模型來建立斷詞候選，並利用混合式 Bi-Gram 分數計算方法，找出最佳的斷詞序列。

第二階段是將第一階段的斷詞結果標記上詞性，使用 2Word-2POS 之詞性模型來建立詞性候選序列，並利用 4-Gram 分數計算方法，找出最佳的詞性標記序列。最後，該論文的斷詞方法，F 分數有 96.69%，

詞性標記方法的 F 分數 92.04%。

3.1.2 文句剖析器

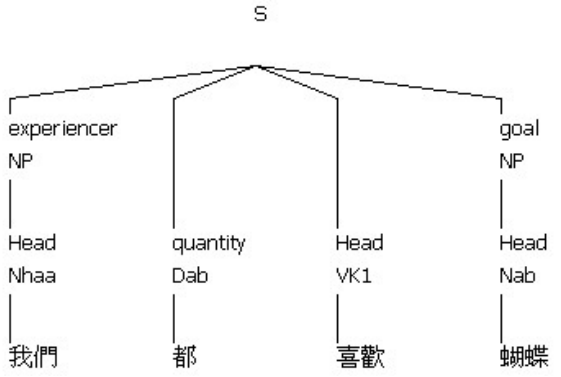
本論文之文句剖析器，主要用於客語斷詞處理結果的剖析。剖析後所得到的文法結構樹，可輸入至韻律階層求取器來得到客語句子中的停頓韻律訊息。接下來我們會介紹該剖析器所使用的語料庫及剖析器之實作方法。

(A) 中文句結構樹資料庫

中文結構樹資料庫(TreeBank)是由 1997 年起由中央研究院詞庫小組(CKIP)從中央研究院現代漢語平衡語料庫(Sinica Corpus)中抽取句子，經由電腦自動剖析成結構樹，再加以人工修正、檢驗後所得的成果。

目前我們所使用的版本為 3.1(Sinica Tree-Bank Version 3.1)，共有約 65434 顆中文剖析結構樹，392237 個詞。這些中文句子的語法結構表達，採取中心語主導原則(Head-Driven Principle)。剖析中文句子時，詞組類型由中心語決定，並且參照中心語和其他成分所記載的語法和語意訊息，表達出句子中詞和詞之間的語法結構和語意角色關係。表五為一顆剖析樹的例子：

表五：中研院剖析樹範例

中文例句	我們都喜歡蝴蝶
結構樹表示法	#S(experiencer:NP(Head:Nhaa:我們)|quantity:Dab:都|Head:VK1:喜歡|goal:NP(Head:Nab:蝴蝶))#
樹結構圖	

符號說明：

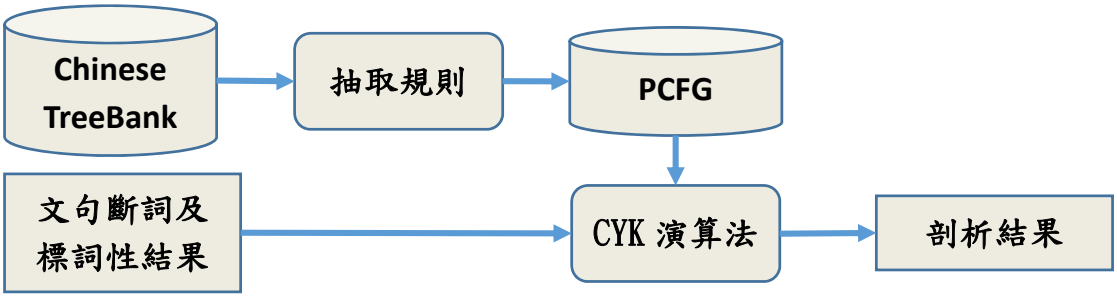
#：以「#」置於前後，作為一結構樹段落。

()：詞組的組合成分為複雜結構，以左括號「**(**」及右括號「**)**」標示其詞組結構的左右邊界。

|：分隔在同一層次上的成分結構。

(B) 文句剖析器實作方法

圖六為文句剖析器實作的示意圖，首先從 Chinese Tree-Bank 抽取出文法生成規則(Production Rules)，並且將生成規則轉換成 CNF(Chomsky normal form)形式、統計出每條生成規則的頻率及機率，建置出 Probability Context-Free Grammar(PCFG)模型。再使用由上而下(Top-Down)的剖析方式及 CYK 動態規劃演算法(Cocke-Younger-Kasami)實作文法剖析器。



圖六：文句剖析器架構圖

3.1.3 韻律階層求取器

本論文的韻律階層求取器，主要是用來找出句子中詞與詞之間較長單位的韻律片段。為什麼我們要預測韻律階層呢？因為人實際在說話時，不會以一個詞為單位來說，而是將整句話分割成多個不同長短的片段來說、有不同的呼吸段，不可能將一句話一氣呵成，或一個一個詞念，有時也會為了要加強語氣或強調重要性，而在不同的時機點加入不同長短的停頓。因此，若在語音合成系統中，預測出適當的韻律片段、加入適當的停頓，也能讓合成出的語音更清楚、更容易理解。

韻律片段的定義，我們採用張唐瑜碩士論文[21]所提出的方法，將詞與詞之間分成三個邊界，分別是：韻律片語(Prosody Phrase)間、韻律詞(Prosody Word)間及韻律詞內。在不同的韻律片段間，我們加入不同程度的停頓，其定義如表六所示：

表六：停頓類型的情況與說明

情況與說明	停頓類型
韻律片語(Prosodic Phrase)間(五大標點之內)	中停頓
韻律詞(Prosodic Word)間(韻律片語之內)	小停頓
詞之間(韻律詞之內)	無停頓
字元之間(詞之內)	連音
在五大標點之間(五大標點“。！？；，”之間)	大停頓

在此舉個實際的例子，假設我們現在輸入的句子是：

這個房間的光線不好不適合做書房。

經過斷詞器的處理後，可以得到結果如：

這 個 房間 的 光線 不 好 不 適合 做 書房。

假設我們直接用斷完詞的結果，詞間不加入任何的停頓來合成出語音。這樣會因為句子太長、念的太快讓人聽不懂。但如果在詞間都加入停頓，則會合成出斷斷續續的語音，讓人聽起來不舒服。如果可以加入適當、少量的停頓，讓整個長句斷成幾個較短的部分，如下：

這個房間的　光線不好 ＊ *不適合　做書房*。

例子中，空白的部分是韻律詞的邊界，即前所述的小停頓(minor break)；而句子下方長線代表韻律片語的範圍，「＊」符號為韻律片語的邊界，就是中停頓(major break)；原先被斷詞器斷開但最後又

被合併的部分，稱為無停頓(no break)。在合成長句的時候，把小停頓和中停頓插入文句中對應的位置，可讓合成語音的語意更加清楚。另外，五大標點符號對應到大停頓，詞內直接連音，這些由斷詞完的結果就可以處理。本模組主要就是預估(predict)詞與詞之間的停頓類型，是中停頓、小停頓或無停頓。

本論文採用蔡育和碩士論文[30]所提出的方法，該論文求取韻律階層的方法，是基於對文法結構樹做掃描(Tree Scan)、合併小樹的過程，以及透過人工標記的韻律階層語料，所統計出的韻律片語(Prosodic Phrase)、韻律詞(Prosodic Word)之字數，來得到韻律階層樹、預測出韻律片語(Prosodic Phrase)及韻律詞(Prosodic Word)之邊界。該論文提出了 2-Pass Scan、Top-Down Scan 及 Bottom-Up Scan 三種方法，其中 2-Pass Scan 方法在主觀評分(Mean Opinion Score, MOS)之聽測實驗，在所有方法中有最佳的分數。

停頓預測，常見的方法是使用統計法，以人工標記出句子的停頓位置後，再抽取出句子的特徵，如詞性、語法類別、停頓類型，透過各種分類器，如決策樹(Classification and Regression Tree, CART)，來統計、訓練出停頓預測的分類模型。這類的方法整體預測正確率較高，但會出現不符合語法結構的錯誤。而本論文所採用的基於文法結構樹來預測的方法，雖然整體正確率較低，但不容易出現不符合語法結構

的錯誤，且與使用決策樹方法做主觀評分聽測實驗的比較，在句子、文章的測驗中，分數都優於使用決策樹的方法。顯見此方法在韻律階層的求取具有不錯的結果，因此本論文採用此方法。

3.2 客語四縣腔語音資料庫

這份語音資料庫為本論文語音合成系統所使用的客語語音合成單元，該資料庫包含了兩大類資料：1.包含所有拼音及聲調的客語單音節音檔，2.客語詞彙音檔。

我們委託熟悉客語四縣腔的 陳婷芳老師(台中市北屯區陳平國小老師)來錄製這些合成單元，其中單音節音檔共有 2427 筆，每個音檔以客語拼音開頭，接上阿拉伯數字聲調做為檔名。詞彙音檔之錄製範本，我們採用「客語能力認證基本詞彙」，共錄製了 2234 筆。另外，我們還錄製了中國大陸地區以及台灣地區的縣市地名音檔共1158筆。這些合成單元音檔錄製格式為：44.1kHz、16 bits，儲存成 Windows PCM 格式(wav 檔)。語音資料庫的資料分佈如表七所示。

表七：2014 興大四縣腔客語語音資料庫分佈表

類型	總數
單音節	2427
詞彙	3392
靜音檔	27
總計	5846

3.3 國客語對照辭典

本辭典用於客語斷辭處理中，找出中文詞之所有可能被轉換的客語詞。本論文所建置的國客語對照辭典，主要來源有：(一)客委會初級[14]、中級暨中高級認證語料[12][13]、(二)台北市客委會-現代客語詞彙彙編、(三) 交大電信工程所，陳信宏老師所提供的-《阿啾箭个故鄉》一書之國客對照詞彙。除了現有的辭典來源外，我們也利用人工標記客語斷詞的同時，加入尚未被收錄在辭典中的詞目。最後，針對每個詞目進行人工校正工作，去除重複或不合理的詞目，得到一部 42772 筆辭目的國客語對照辭典。辭典的資料樣貌及資料分佈，如表八及表九所示。

表八：2014 興大國客語對照辭典資料樣貌

欄位	內容
國語詞	年輕人
客語詞	後生人
詞性	Na

表九：2014 興大國客語對照辭典分佈統計表

字詞	總數
一字詞	4302
二字詞	25716
三字詞	6851
四字詞	5033
五字詞	503
六字詞	214
七字詞	125
八字詞	28
總計	42772

3.4 客語發音辭典

發音辭典對於一個語音合成系統來說非常重要，其用途是求句子中每個字的發音，再合成出語音。本論文客語發音辭典，主要來源為：(一)客委會初級[14]、中級暨中高級認證語料[12][13]、(二)台北市客委會-現代客語詞彙彙編。我們將發音辭典分為兩部分，1.客語詞彙發音辭典(表十)，共 32453 筆發音資料(表十一)、2.客語單音節發音辭典(表十二)，共 9362 筆發音資料。

表十：客語詞彙發音辭典資料樣貌

欄位	內容
客語詞彙	發熱痱仔
客語發音	bod4　ngied1　bi1　e4
詞性	VH

表十一：客語詞彙發音辭典分佈統計表

字詞	總數
一字詞	2175
二字詞	19190
三字詞	6685
四字詞	3947
五字詞	301
六字詞	81
七字詞	65
八字詞	9
總計	32453

表十二：客語單音節發音辭典資料樣貌

欄位	內容
客語字	雙
客語發音	sung2

3.5 客語句子平行語料

本客語句子平行語料用於建置客語語言模型，我們將在第四章詳細介紹如何處理該語料，以及建置客語語言模型。本語料來源是客委會四縣腔初級[14]、中高級客語認證教材[12][13]。這份語料，句子數分別有初級 1678 句、中高級 4962 句，共 6640 句，每一句都有中文、客語的詞目、拼音及例句，資料格式表十三：

表十三：客委會認證教材語料的資料格式

句子編號	01-001
客語詞	光線
中文詞	光線
客語拼音	gong∨ sien
中文例句	這個房間的光線不好，不適合做書房。
客語例句	這隻房間个光線毋好，毋適合做書房。

第四章 國客語對應式語言模型建置

本章節將詳細說明客語語料的處理方法，以及客語語言模型的建置方式。我們將介紹客語斷詞標記工具，以及客語斷詞的標記原則。並且會詳細說明語言模型的基礎理論，以及如何建置國客語對應式的語言模型。

4.1 客語句子平行語料的處理

目前電腦上的客語語料仍非常缺乏，能用來建置語言模型的語料十分有限。而若要建置客語語言模型，還必須先有完成客語斷詞標記結果的語料，才能進行語言模型的建置工作。本小節即是在介紹我們如何建置客語斷詞語料。

因為客文和中文都屬於華文，兩者的文法結構相近，因此中文斷詞和詞性的標記的結果，大部分都能與客語對應，僅有少部分的客語俚語或特殊用詞例外。而中文語料的處理，因目前中文斷詞系統的發展已相當成熟，因為中文語料的龐大，以規則法配合機率模型的混合式斷詞法所發展出來的中文斷詞系統，斷詞效能及詞性標記的正確率已達到九成五及九成二以上。因此都能直接以中文斷詞系統來得到可靠的斷詞及詞性特徵標記的結果。但客語語料的處理，因目前市面上及學界的客語斷詞系統仍處於發展中的階段，還沒辦法完全依賴任何

客語斷詞系統來自動處理、標出斷詞及詞性標記結果，因此目前都必須透過人工方式去標記。

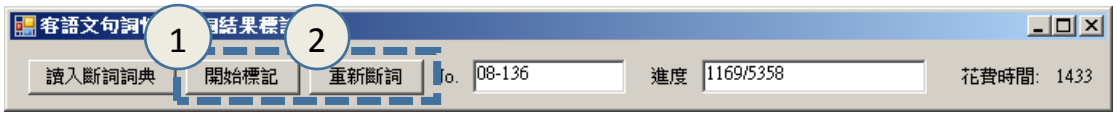
為了快速標記出客語斷詞語料，我們開發一個半自動式的客語斷詞標記工具，來處理客語平行語料。透過這個工具，可以快速的將國客語對應式句子語料，標記出客語斷詞及詞性結果。我們再將該結果統計出客語 Uni-gram 及 Bi-gram 詞頻，並建置成對應式客語語言模型，做為客語斷詞處理中的語言模型。

4.1.1 客語斷詞標記工具介紹

圖七為本工具之介面，接下來我們會詳細介紹各區塊的功能以及操作方式。

圖七：客語斷詞標記工具畫面

(A) 開始標記、重新斷詞



圖八：客語斷詞標記工具「開始標記、重新斷詞」畫面

1. **開始標記**：開始標記作業，若對當下的標記結果不滿意，也可按此按鈕重置該句標記結果。
2. **重新斷詞**：在人工修改中文句子後，按重新斷詞後，可得到新的中文斷詞「候選序列表」。

(B) 中文句子文字框、詞性敘述、該詞可能詞性

圖九：客語斷詞標記工具「中文句子文字框、詞性敘述、該詞可能詞性」畫面

1. 中文句子文字框

a. 顯示目前標記目標的中文句子。

b. 在中文句子字串上，以[滑鼠左鍵]反白一段字串，可查詢該字串所能構成的詞有哪些，以及它們的詞性，以供增加候選詞。如圖，在「運動」上反白後，可查出該字串能構成的詞，以及每個詞的詞性。

2. 詞性敘述：說明「**該詞可能詞性**」區域中，所有的詞之詞性的意義，以輔助標記者做出較正確的選擇。

3. 該詞可能詞性：除了上述的功能外，這個區域中，還能做以下的操作與功能：

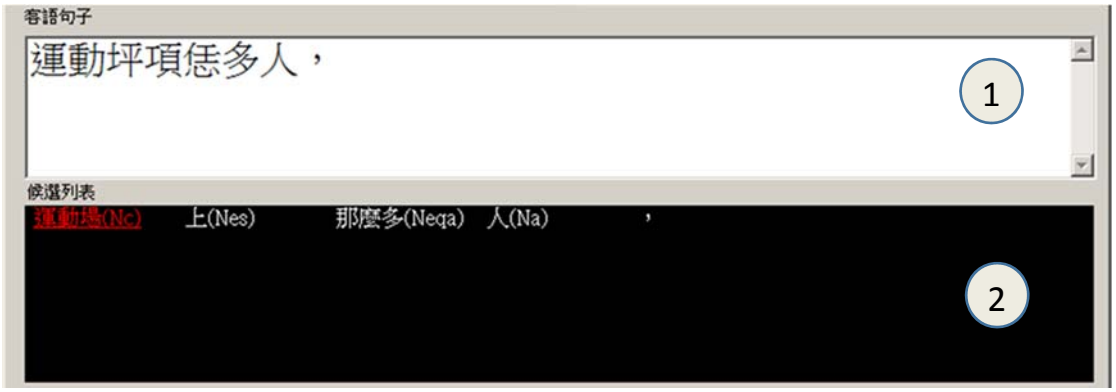
a. 當標記者在「**中文句子文字框**」中以[滑鼠左鍵]反白一段字串時，

依照上述的功能，可查詢該字串可構成的詞及其詞性，並顯示在此區域中。但是若發生查不到的情況，這時該字串會被當成是客語用詞，並給予詞性(HK)，給予標記者創造特殊客語用詞的彈性。

b. 在該窗的詞上按右鍵，可將該詞加入到「**候選列表**」變成候選詞。

以上敘述的這些組合功能可供標記者修正錯誤的詞性標記或斷詞結果。

(C) 客語句子、候選列表



圖十：客語斷詞標記工具「客語句子、候選列表」畫面

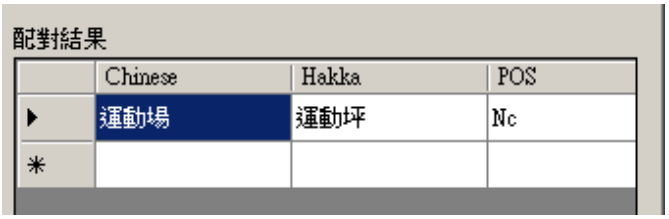
1. 客語句子文字框：

a. 顯示目前標記目標的客語句子

b. 在客語句子上以[滑鼠左鍵]反白一段字串，依照不同狀態，可進行以下操作：

甲、狀態 1：在「候選列表」窗中未有鎖定目標時(所有詞皆無底線之狀態)，在文字框上反白一段字串，可查詢該字串可組成的「中文詞及其詞性」，以供標記者增加候選詞（顯示在候選列表中）。

乙、狀態 2：在「候選列表」有鎖定目標時(有底線者)，在文字框上反白一段字串，可將表中被鎖定的中文詞與該字串配對成一個國客語對應詞，配對結果會顯示在「對應結果」框中，如下圖所示。



2. 候選列表：

a. 顯示中文斷詞結果，並變成候選詞讓標記者挑選。

b. 該窗還可進行以下的操作：

甲、可按[右鍵]來*選擇*或*取消*該詞的鎖定狀態，被鎖定的詞會呈現紅色字體且有底線，被取消的詞會呈現白色字體且沒有底線。如圖：

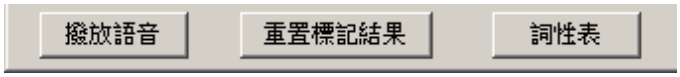
鎖定「運動場(Nc)」之畫面

取消鎖定「運動場(Nc)」之畫面

乙、在詞上按[滑鼠左鍵]，能刪除該詞。如圖：

在「運動場(Nc)」上按滑鼠左鍵，刪除該詞。

(D)播放語音、詞性表

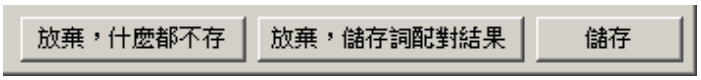


圖十一：客語斷詞標記工具「播放語音、詞性表」畫面

1. **撥放語音**：可撥放該客語例句的與音檔案，以輔助標記作業。(有時可從念的斷點找出詞邊界)

2. **詞性表**：讓標記者查詢所有的詞性。

(E) 儲存結果



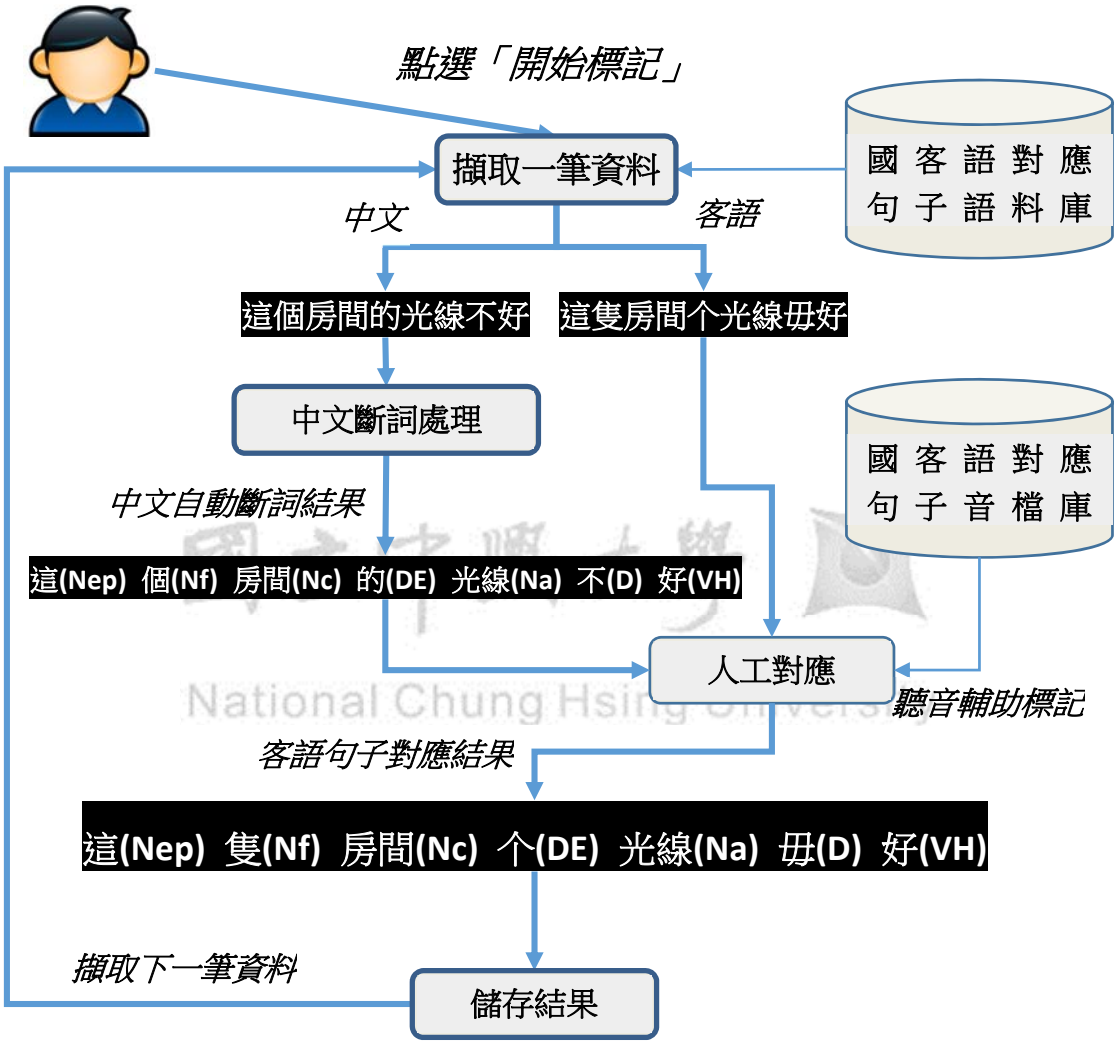
圖十二：客語斷詞標記工具「儲存結果」畫面

1. **放棄、什麼都不存**：此次的標記，難到不知道怎麼對應，連一個詞都找不到，只能完全放棄儲存。

2. **存詞配對結果**：此次的對應雖無法對應完所有客語詞，但有找到一些合理的對應，所以只存對應後的詞。

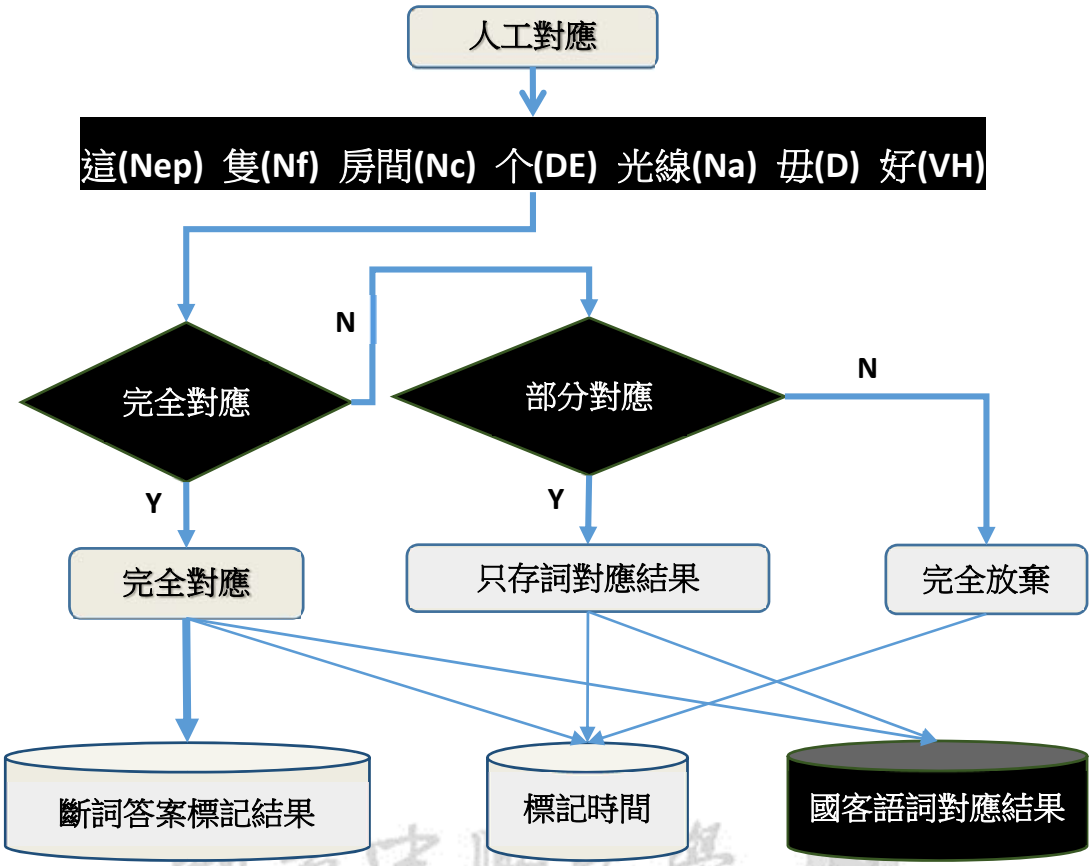
3. **儲存**：此次標記完美對應，儲存標記好的結果。

在此我們舉個標記一筆語料的例子，來說明整個標記的過程。圖十三為標記一筆「這個房間的光線不好/這隻房間个光線毋好」語料的示意圖。



圖十三：客語斷詞標記工具程式運作程圖

透過上述工具所提供的各種功能，標記者在「人工對應」標記時，選出最適當的對應結果，並儲存標記結果至資料庫中。下圖是完整的資料儲存流程。



圖十四：客語斷詞標記工具之標記流程圖

4.1.2 客語斷詞標記工具使用結果討論

我們將客委會四縣腔認證教材的客語語料 6640 句句子，依照句子本身的編號順序，由小到大分為語料 A 及語料 B，語料 A 有 4196 句，語料 B 有 2444 句，並再將 B 語料依編號順位分為 4 份，B1-B4，每份 611 句。

將 A、B 語料，分別找(標記者 1)一位碩士生、(標記者 2~5)四位大學在學生，使用本論文所開發的客語斷詞標記工具，以半自動人工判

斷方式，標記出客語斷詞的結果。其中每人的標記所花費時間如表十四：

表十四：客語斷詞標記工具的標記時間統計，時間單位：秒

標記者	1	2	3	4	5	總平均
語料編號	A	B1	B2	B3	B4	
處理句數[4]	4500	615	622	611	646	
儲存句數	4018	346	451	504	419	
總時間[5]	91798	28698	25297	18864	17219	
平均每句	20.39	46.66	40.67	30.87	26.65	33.04

表十四中顯示，標記速度平均 33.04 秒能完成一句。而標記完成的資料，會儲存其 1.中文句子、2.中文斷詞及詞性標記結果、3.客語句子、4.客語斷詞及詞性標記結果、5.客委會語料中的句子編號以及 6.標記時間，等六個欄位資料，儲存結果如下表：

表十五：客語斷詞標記結果的資料樣貌

中文句子	這個房間的光線不好
中文斷詞結果	這(Nep) 個(Nf) 房間(Nc) 的(DE) 光線(Na) 不(D) 好(VH)
客語句子	這隻房間个光線毋好
客語斷詞結果	這(Nep) 隻(Nf) 房間(Nc) 个(DE) 光線(Na) 毋(D) 好(VH)
句子編號	01-001
標記時間	21

[4] 處理句數的統計，包含重新處理曾經放棄的句子，因此實際處理可能會比分配到的筆數多。
[5] 時間單位為秒。

我們將這些標記完成的資料再經過一次經人工篩選後，最後確認有效句數為：A 語料 4018 句，B1-B4 語料共 1282 句，我們使用語料的分佈如表所示：

表十六：客語語料的使用分佈

	訓練	測試
句數	4018	1282
詞數	45304	17646
字數	65572	25478

4.1.3 客語斷詞標記原則

專家們編審這份客語認證教材語料的主要目的是客語教學用途，並不是要建立「國語/客語斷詞對應」的語料，所以在撰寫例句時並不會特別強求或注意到國客語的完全對應。

因此，我們在進行人工標記時也發現，其實大部分出現無法對應的情況，都可以以人工修改中文句子或用詞的方式，做適當的修飾與調整，來達到不影響文意、又能與客語句子對應完全的目的。但某些句子仍無法確認如何標記時，標記者也可選擇放棄該句的標記，或只存能對應到的詞，而不將這些未完成對應的句子視為斷詞答案。我們將這些情況歸納出五大標記原則，以下我們會詳記的介紹這些原則的情況與處理方法。

(A)原則一：以不修改客語句子結構及原意為原則，但可跳過或增加不影響文意的字。

表十七：標記原則一之例句一

中文	這泉水的水質很甜很清澈
客語	這**窟**泉水个水質當甜當清
客語改	這泉水个水質當甜當清

此例子中，「窟/fud`/」字通常跟盛液體的容器或凹槽、坑洞有關，例如我們常見的用法：「酒窟」、「水窟仔」、「窟仔」。但在這裡用在「泉水」這個詞前面，只是要形容「泉水」在水坑裡的意思，並非與「泉水」組成一個詞來使用。加上該句語料的中文句子中，並沒有「窟」字，我們無法完全將客語句子與中文句子的每個字詞做對應。但我們發現省略該字後，也不影響句子要表達的意思。因此遇到此類的情況，我們可透過這個原則做處理，讓句子能順利標記完成。下面是這句客語修改後，與中文完全對應的結果。

表十八：標記原則一例句的斷詞標記結果

中文	這(Nep) 泉水(Na) 的(DE) 水質(Na) 很(Dfa) 甜(VH) 很(Dfa) 清澈(VH)
客語	這(Nep) 泉水(Na) 个(DE) 水質(Na) 當(Dfa) 甜(VH) 當(Dfa) 清(VH)

表十九：標記原則一之例句二

原始 中文例句	太陽下山以後就***可以看得到***滿天的星星
原始 客語例句	日頭落山以後就***看得著***滿天个星仔
修改 中文例句	太陽下山以後就***看得到***滿天的星星

此例子中，可發現原始的中文例句中，多了「可以」這個詞，很明顯的這個詞在原始客語例句是沒有的。這種情況的處理方法有兩種，第一是將中文句子中的「可以」直接省略，第二是在客語例句中，加入「做得/zo ded`/」。

(B)原則二：有明顯的斷詞錯誤，要人工介入修正。

表二十：標記原則二之例句

原始 中文斷詞	每(Nes) ***天都(Na)*** 在(P) 沙洲(Na) 上(Nes) 玩(VC) 摔跤(VA)
修正 中文斷詞	每(Nes) ***天(Nf) 都(Da)*** 在(P) 沙洲(Na) 上(Nes) 玩(VC) 摔跤(VA)

此例子中，「天都/ㄉㄨ/」一詞，被誤斷成一個名詞 Na，這跟「天(Nf)、都(Da)/ㄉㄡ/」指「每天都如何」的意思有天壤之別。「天都」是地方名稱，指的是古代上天帝王居住的首都。但我們這裡指並非指天上的首都，這個自動斷詞結果明顯有嚴重的錯誤，若不修改此類錯誤，會直接影響到該詞的念法。而其正確斷詞應該斷成「天(Nf) 都(Da)」。因此遇到此類情況，我們都必須人工修正

自動斷詞結果，再進行國/客斷詞對應的標記。

(C)原則三：以不修改客語句子結構及原意為原則，可微調中文句子用詞及詞的順序，以求能對應到客語。

本原則與原則一有雷同之處，但為了更確立標記的標準，我們定出這個原則，授權標記者能對句子做適當的修改。

表二十一：標記原則三之例句

原始中文例句	古時候的人會觀察***天上的星宿變化***
原始客語例句	上早个人會觀察***天頂星宿个變化***
修改中文例句	古時候的人會觀察***天上星宿的變化***

此例子中，客語例句中「天頂星宿个變化」與中文「天上的星宿變化」雖然詞的詞順序不同，但中文句子中的「的」調換後，也不影響文意。

修改句子的動作，我們只建議修改中文，客文若非必要儘量保持原句。原因是較能保持客語句子原來的特性、結構、用語…等資訊。

(D)原則四：發現中文詞和客語詞的對應有爭議或太過模糊，要找過一個最佳選擇的詞替換。

表二十二：標記原則四之例句

原始中文例句	會想到***一些***壞兆頭
原始客語例句	會想著***麼个***壞兆頭
修改中文例句	會想到***什麼***壞兆頭

在此例子中，客語的「麼个/ma` ge/」，中文是「什麼」的意思。在有些情況下是用於疑問句，例如：「你喊麼个名？」，意思是問「你叫什麼名字？」。但在有些情況並非真的要問什麼，例如：「麼个人講麼个話」，意思是說「什麼人講什麼話」。在本例句的情況，是屬於後者。

我們可以由中文例句中發現，將「麼个」翻成中文「一些」，依文意來看過於模糊、不精準，直接改用「麼个/什麼」的對應也並不會影響意思。因此，可透標記工具中的詞典查詢功能，找到客語詞「麼个」能翻成的國語詞有哪些，發現到「什麼」這個中文詞最貼近文意。因此，手動將中文句子的「一些」改為「什麼」，重新與「麼个」配對。

(E) 原則五：配對時以詞為單位，不能以片語為單位。

表二十三：標記原則五之例句一

原始中文例句	現在外面***既颱風又下雨***
原始客語例句	這下外背***風合雨***
原始中文例句斷詞結果	現在(Nd)　外面(Ncd)　既(Caa)　颱風(VA)　又(D)　下雨(VA)

此句語料中，我們發現「既颱風又下雨」和「風合雨」看似能對應，但實際上，這樣的用法可能只是偶然出現。且「既颱風又下雨」是形容又颱風又下雨的情況，並非單一的詞。因此，這種非成語的片語，我們不儲存此類的答案和詞配對。但「現在/這下、外面/外背」等詞，皆能確定其對應，我們可將這些詞的國客語對應結果儲存起來，待後續的工作中仍可應用。

表二十四：標記原則五之例句二

原始中文例句	一身***乾彆簡單粗陋的衣服***都沒有能力買
原始客語例句	一身***腊食皮***無才調買
原始中文例句斷詞結果	一(Neu)　身(Na)　乾彆(Na)　簡單(VH)　粗陋(VH)　的(DE)　衣服(Na)　都(D)　沒有(VJ)　能力(Na)　買(VC)

此句語料中，我們發現除了「一/一、身/身、無/沒有、才調/能力、買/買」這些詞能確定其國客語詞的對應外，其中「乾彆簡

單粗陋的衣服」雖然看似能對應到「腊食皮」，但這樣的用法也許只是偶然出現，且「乾彆簡單粗陋的衣服」是形容一件極簡陋的衣服，並非單一的詞。因此我們不儲存這類的結果。但有部分詞能確定其對應，我們可將這些詞的國客語對應結果儲存起來，日後可增加到國客語對照辭典。

4.2 語言模型介紹

4.2.1 語言模型的概念

語言模型(Language Model)是斷詞演算法中，用來選擇出最佳斷詞詞序列的重要元件。目前語言模型的設計，可分為：(1)以文法取向(2)以統計取向，兩種設計方法[16][27]。

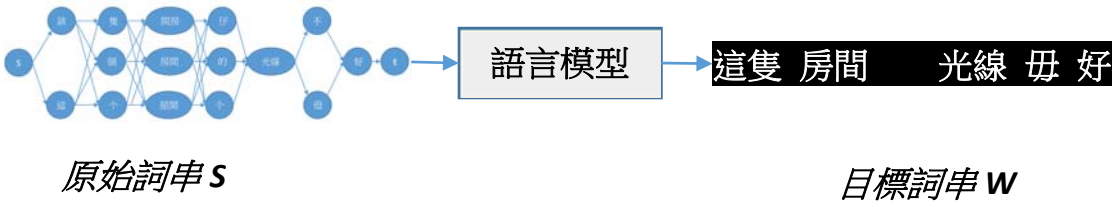
(1) 文法取向：

這類語言模型的作法，是根據文法[19]及語意規則[26]來制定條件及規則。再經由文句剖析器(Parser)對句子做剖析，得到文法結構樹、建置成文法取向的語言模型。其優點是易於擷取詞彙或文句中的意義，可用於語言處理，如：機器翻譯、斷詞處理…等。但缺點是應用在語音辨識時，無法處理語法不合句法規則的句子。這類語言模型能應用的範圍有限，因此目前的語言模型，都還是以統計取向為主。

(2) 統計取向：

這類語言模型的作法，是透過大量的訓練語料，統計出該語料的機率分佈模型。若使用在多個可能斷詞序列的選擇上，是將長度為 N 的詞串(N-Gram)機率，從語言模型中查出，並且找出可組成最大機率的詞串序列。而應用於語音辨識上，該方法的優點是一定能辨識出一個最可能的句子，對於語法錯誤的容錯率高，但缺點是必須收集大量的訓練語料進行訓練。

如圖十五所示，一個語言模型被應用在決定最佳斷詞序列(找出一個機率最大的目標詞串W)的例子。原始詞串S中有多個可能詞串序列，透過語言模型計算機率後，將得到最佳的目標詞串W，註記成$w_1, w_2, w_3, \ldots, w_m$或是$w_1^m$，$w_i \in V$，代表$m$個詞所組成的詞串，$V$則是詞典為所有詞的集合。$S$可以為任何所有可能的詞串。



圖十五：語言模型在客語斷詞處理的應用

一個目標詞串 $W = w_1 \ldots w_m$ 的機率 $P(W)$ ，W 為 $w_1, w_2, \cdots, w_m$ 是一個欲轉換的字串 S ，可表示如下：

$$P(w_1 w_2 w_3 \ldots w_m) \tag{4-1}$$

上式(4-1)可以寫成：

$$P(w_1^m) = P(w_1)P(w_2 \mid w_1^1)P(w_3 \mid w_1^2)\ldots P(w_m \mid w_1^{m-1})$$

$$= P(w_1)\prod_{i=2}^{m} P(w_i \mid w_1^{i-1}) \tag{4-2}$$

其中 $P(w_i \mid w_i^{i-1})$ 是詞 w_i 在特定歷史詞串 $h_i = w_1, w_2, w_{3,} w_4, \cdots, w_{i-1}$ 的情況下，出現的條件機率。實際上在建立語言模型時，並不會把所有可能的參數 $P(w_i \mid w_1 w_2 w_3 \ldots w_{i-1})$ 都儲存起來。因為針對長度 m ，歷史詞串長度為 $n-1$ 時，所有可能的組合個數為 $|V|^n$ 。這樣的情況下，即使詞典所收錄的詞彙量不大，但只要詞串長度稍長，參數就會有驚人的成長，因此必須對參數加以簡化。譬如歷史詞串長度 $n-1=0$ 時，可以表示如下：

$$P(w_1^m) = \prod_{i=1}^{m} P(w_i) \qquad \text{(uni-gram)} \tag{4-3}$$

歷史詞串長度 $n-1=1$ 時，可以表示如下：

$$P(w_1^m) = P(w_1)\prod_{i=2}^{m} P(w_i \mid w_{i-1}) \qquad \text{(bi-gram)} \tag{4-4}$$

歷史長度$n-1=2$時，可以表示如下：

$$P(w_1^m) = P(w_1)P(w_2 \mid w_1)\prod_{i=3}^{m} P(w_i \mid w_{i-2}^{i-1}) \qquad \text{(tri-gram)} \qquad (4\text{-}5)$$

通式則可以表示如下：

$$P(w_1^m) = \prod_{i=1}^{m} P(w_i \mid w_{i-n+1}^{i-1}) \qquad \text{(n-gram)} \qquad (4\text{-}6)$$

從統計觀點估測$P(w_i \mid w_{i-n+1}, w_{i-n+2}, ..., w_{i-1})$的方式，是根據最大相似度估測法(Maximum likelihood estimation, MLE)，得到下式:

$$P(w_i \mid w_{i-n+1}, w_{i-n+2}, ...w_{i-1}) = \frac{C(w_{i-n+1}, w_{i-n+2}, ..., w_i)}{C(w_{i-n+1}, w_{i-n+2}, ..., w_{i-1})} \qquad (4\text{-}7)$$

其中$C(\bullet)$表示詞串出現的次數。

4.2.2 語言模型的評估

本論文將使用交叉熵(Cross Entropy)和混淆度(Perplexity)，來評估語言模型的效能，並透過實驗觀察出最佳的語言模型平滑方案。交叉熵和混淆度是相當重要且通用的標準，混淆度是根據資訊理論(Information Theory)而來。對一組測試資料T，其中$e_1,e_2,\cdots,e_m$為m個測試事件，則測試語句T的機率可以表示如下:

$$P(T)=\prod_{i=1}^{m}P(e_i) \tag{4-8}$$

其中$P(e_i)$為每個 events 的機率值，在測試資料中$H(T)$可視為這m個 events 需要編碼的長度位元數，表示如下:

$$H(T)=-\sum_{i=1}^{m}P(e_i)\log_2 P(e_i) \tag{4-9}$$

$$PP(T)=2^{H(T)} \tag{4-10}$$

整體而言，較低的$H(T)$可以推導出較低的$PP(T)$，意即擁有較低$PP(T)$的語言模型，會有較好的性能。因此混淆度可以解釋成語言模型估測一個詞串後面平均可能的可接詞數。混淆度低，表示一個詞串後面有較少的選擇，辨認時就愈能找到正確的答案。

另外，交叉熵CH亦是一種量測語言模型的方式，若是語言模型能夠精準的預測出接下來的 events，則交叉熵勢必較低。在一般的情

況下，$CH >= H$ ，H 表示使用相同的模型進行訓練、測試。實際上，我們對測試模型的機率分佈並不清楚，所以必須依靠訓練模型 M 進行預估。M 的交叉熵表示如下:

$$CH(P,M) = -\sum P(e)\log_2 M(e) \tag{4-11}$$

根據 Shannon-McMillan-Breiman theorem0[6]可以化簡為如下:

$$CH(P,M) = \lim_{n\to\infty}\frac{1}{n}\log_2 M(w_1 w_2 w_3 ... w_n) \tag{4-12}$$

交叉熵 CH 是熵 $H(p)$ 的上限值，換句話說，$H(P) <= CH(P,M)$ ，其意義指的是可以用訓練模型 M 來估計每個 events 的機率。

4.2.3 語言模型的平滑化方法

由於語言模型的訓練語料中，每個 events 之間存在一定的局限性[20]和片面性，許多合理的 events 搭配沒有出現在訓練語料中。例如：一個 Bi-Gram 詞串 $Event(W_{i-1}, W_i)$ 沒有在訓練語料中出現，根據式(4-6)，該詞串對應的上下文條件機率 $P(W_i|W_{i-1}) = 0$ 從而導致該詞串所在的句子出現機率為零，這種情況通常被稱為資料稀疏或零機率問題，而平滑化即是要針對此問題進行解決。

本論文採用(1)加成平滑法(Additive Smoothing Method)、(3)凱氏平滑法(Katz smoothing Method)，以及 Back-off 架構的(4)強化凱氏平滑法

(Enhanced Katz Smoothing Method)[16]兩個方法來解決料稀疏問題，並透過實驗找出適切的解決方案。

(1) 加成平滑法(Additive Smoothing Method)：

加成平滑法是一個很基本且直觀的方法，由 Lidstone、Johnson 和 Jeffreys 等人提出了一種簡單易行的數據平滑方法[4]。它的基本思想是為了避免零機率的問題，將語言模型中每個 events 的出現次數加上一個常數 δ。它的做法是將一個常數 δ 加到所有的 events 中(包含所有已出現過的 seen events 和未出現過的 unseen events)。即是原本一個出現 c 次的 event 加上 δ 次後調整成為新的出現次數 c^*，但是調整以後全部 events 的次數還要維持 N。

式子表示如下:

$$c^* = (c+\delta)\frac{N}{N+V\delta} \tag{4-13}$$

V 代表辭典中所有詞彙的個數， N 代表資料(events)的數目。

其機率值為:

$$Q^*_{c,N} = \frac{c^*}{N} = \frac{(c+\delta)}{N+V\delta} \tag{4-14}$$

一般而言，0< δ <=1，在 δ =1 的情況下被稱為加 1 平滑法。在這樣的情況下，c 次被調整後的次數 $c^* = (c+1)\frac{N}{N+V}$ for $i \geq 0$ 而 $\frac{N}{N+V}$ 是對出現 c^* 次的 events 進行調整的正規化係數(Normalization Factor)。

(2) 古德圖靈平滑法(Good-Turing Smoothing Method)：

Good-Turing 平滑法[3]於 1953 年第一次由 Good 提出，亦有一些相關研究[2][5][7]，是許多平滑技術的核心。Good-Turing 估計的基本思想為：對於 N-gram 語言模型中出現c次的 events w_{i-N+1}^{i}，根據 Good-Turing 估計公式，該 events 的出現次數為c_{GT}^{*}:

$$c_{GT}^{*} = (\mathrm{c}+1)\frac{n_{c+1}}{n_c} \tag{4-15}$$

而n_c代表的是語言模型中恰好出現c次的 events 共有幾個，例如：n_0代表的是在這個語言模型中，恰好出現零次的 events 數目，即沒出現過的 events 數目; n_1代表的是在這個語言模型中，恰好出現一次的 events 數目。因此，n_c可表示如下：

$$n_c = \sum_{w:C(w)=c} 1 \tag{4-16}$$

其機率值為:

$$Q_{c,N}^{*} = \frac{c_{GT}^{*}}{N} \tag{4-17}$$

由於n_c不能為零，所以 Gale 和 Sampson(1995)提出了一種用於 n_c 的平滑算法 Simple Good-Turing[8]。因為 Good-Truing 估計公式中缺乏利用低元模型對高元模型進行線性插值的思想，所以它通常不單獨使用，而是作為其它平滑算法中的一個計算工具。

(3) 凱氏平滑法 (Katz Smoothing Method)：

在 Good-Turing 平滑法中並沒有提到該如何處理$n_{c+1}=0$的情況。Katz(1987)提出一個計算c^*的方法：

$$c_{katz}^{*}=\frac{(c+1)\frac{n_{c+1}}{n_c}-c\frac{(k+1)n_{k+1}}{n_1}}{1-\frac{(k+1)n_{k+1}}{n_1}} \quad ,for\ 1\le c\le k \tag{4-18}$$

並設立門檻值k，只對出現次數為 1 到k次的 events 做調整，若出現次數比k還要大的 events 不做調整，而 Katz 建議k值為 5。

$$c_{katz}^{*}=c \quad ,for\ c>k \tag{4-19}$$

凱氏平滑法是以 Back-off 為架構的平滑方法，這個架構的概念，是先將在訓練語料中出現的已知事件(Seen Events)的機率，套上一個折扣函數(Discount Function) $d(c)$，折扣出小量的機率。再將折扣出的機率量依照 Back-off 的分佈(Back-off Distribution)，分給不曾出現在訓練語料中的未知事件(Unseen Events)，來避免出現零機率的情況。公式(4-20)為應用於 Bi-gram 模型的凱氏平滑法公式：

$$P_{katz}(W_i|W_{i-1})=\begin{cases}\frac{C(W_{i-1},W_i)}{\sum_{W_k}C(W_{i-1},W_k)} & if\ C(W_{i-1},W_i)>k\\ d(c)\times\frac{C(W_{i-1},W_i)}{\sum_{W_k}C(W_{i-1},W_k)} & if\ 0<C(W_{i-1},W_i)\le k\\ \alpha(W_{i-1})\times\frac{C(W_i)}{\sum_{W_j:C(W_{i-1},W_j)=0}C(W_j)} & if\ C(W_{i-1},W_i)=0\end{cases} \tag{4-20}$$

其中$d(c)$ $(0<d(c)\le 1)$為 Bi-Gram 模型之n_c分佈的折扣函數，是透過公

示(4-15)的 Good-Turing Estimation 及公式(4-18) Katz 公式求得，計算方式如公式(4-21)。$\sum_{W_j:C(W_{i-1},W_j)=0} C(W_j)$為 Back-off 後的分佈，而$\alpha(W_{i-1})$是一個正規化係數(Normalization Factor)，也就是由訓練語料出現次數為$1 \leq c \leq k$的已知事件(Seen Events)所折扣出來的機率量，計算方式如公式(4-22)。

$$d(c) = \frac{c^*_{katz}}{c} \quad (4\text{-}21)$$

$$\alpha(W_{i-1}) = 1 - \sum_{W_j:C(W_{i-1},W_j)>0} P_{katz}(W_j|W_{i-1}) \quad (4\text{-}22)$$

而公式(4-20)可這樣解釋，當$C(W_{i-1}, W_i) > k$時，表示該事件的出現次數夠大、機率值非常可靠，不需要折扣出任何機率，因此$d(c) = 1$。

當$0 < C(W_{i-1}, W_i) \leq k$時，則我們給予一個小於 1 的折扣值，折扣出機率再分給 back-off 後的分佈。也就是當$C(W_{i-1}, W_i) = 0$時，依照$C(W_i)/\sum_{W_j:C(W_{i-1},W_j)=0} C(W_j)$之比例來得到折扣出的機率值。

(4) 強化凱氏平滑法 (Enhanced Katz Smoothing Method)：

強化凱氏平滑法，是基於凱氏平滑法的基礎而發展的，呂宜玲碩士論文[16]所提出。該論文主張一般的凱氏平滑法並未處理當$C(W_i)=0$時的情況，當$C(W_i)=0$時會造成零機率問題。因此提出了強化凱氏平滑法來解決此問題。公式(4-22)為應用於 Bi-Gram 模型的強化凱氏平滑法：

$$P_{Ekatz}(W_i|W_{i-1})=\begin{cases}\frac{C(W_{i-1},W_i)}{\sum_{W_k}C(W_{i-1},W_k)} & if\ C(W_{i-1},W_i)>k\\ d(c)\times\frac{C(W_{i-1},W_i)}{\sum_{W_k}C(W_{i-1},W_k)} & if\ 0<C(W_{i-1},W_i)\le k\\ \alpha(W_{i-1})\times\frac{C(W_i)}{\sum_{W_j:C(W_{i-1},W_j)=0}C(W_j)} & if\ C(W_{i-1},W_i)=0\ and\ C(W_i)>k\\ d'(c)\times\alpha(W_{i-1})\times\frac{C(W_i)}{\sum_{W_j:C(W_{i-1},W_j)=0}C(W_j)} & if\ C(W_{i-1},W_i)=0\ and\ 0<C(W_i)\le k\\ \beta(W_{i-1})\times\frac{1}{T} & if\ C(W_{i-1},W_i)=0\ and\ C(W_i)=0\end{cases} \quad (4\text{-}22)$$

其中$d'(c)$ $(0<d'(c)\le 1)$為 Uni-Gram 模型之n_c分佈的折扣函數，與$d(c)$為 Bi-Gram 模型之n_c分佈不同。$d'(c)$是透過 Uni-Gram 模型的n_c之分佈所求出，計算方式如公式(4-23)及公式(4-24)。

$$c'^{*}_{katz}=\frac{(c+1)\frac{n'_{c+1}}{n'_c}-c\frac{(k+1)n'_{k+1}}{n'_1}}{1-\frac{(k+1)n'_{k+1}}{n'_1}}\ ,for\ 1\le c\le k \quad (4\text{-}23)$$

其中n'_c為 Uni-Gram 模型之恰好出現c次的 Events 之個數。

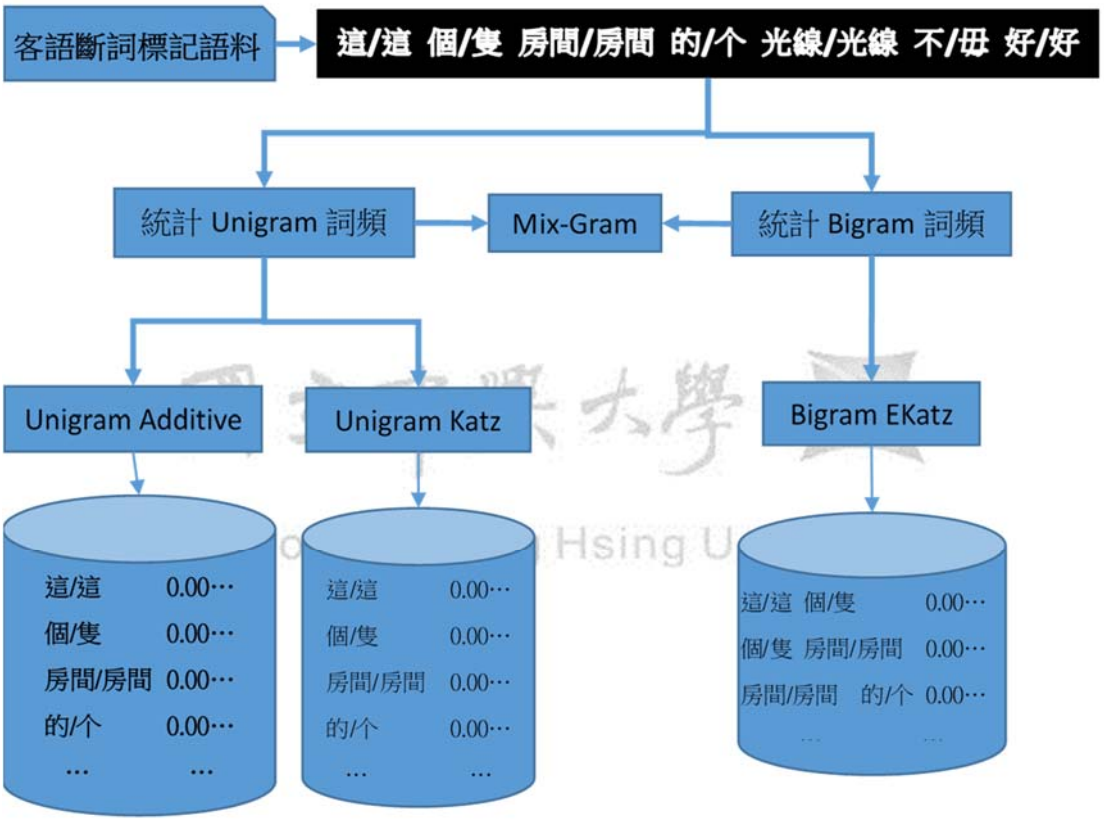
$$d'(c)=\frac{c'^{*}_{katz}}{c} \quad (4\text{-}24)$$

而$\beta(W_{i-1})$為未與W_{i-1}相鄰出現，但在訓練語料中出現，且其次數c在$1\le c\le k$者，所折扣出的機率量，計算方式如(4-25)。而T為辭典中未出現在訓練語料的詞之個數。

$$\beta(W_{i-1}) = \alpha(W_{i-1}) - \sum_{W_j:C(W_{i-1},W_j)=0\ and\ C(W_j)>0} P_{katz}(W_j|W_{i-1}) \quad (4\text{-}25)$$

4.3 國客語對應式語言模型的建置

本節將介紹本論文建置語言模型的方法，圖十六為本論文所建置的語言模型之示意圖。



圖十六：國客語對應式語言模型建置結果示意圖

我們分別建置了 Uni-Gram 及 Bi-Gram 語言模型，其關係如圖十六所示。其中 Uni-Gram 語言模型，我們使用了加成平滑法及凱氏平滑法做語言模型的平滑處理。而 Bi-Gram 語言模型，我們使用加成平滑法及強化凱氏平滑法做平滑處理。最後，透過實驗觀察，我們提出了結

合兩個語言模型的 Mix-Gram 方法，但是此模型並不符合機率之基本定義，因此我們稱為 Mix-Gram 分數(詳見 5.9 節)。

一般語言模型的訓練方式，是以一個詞為單位，如下列句子(空白處代表斷詞邊界)：

中文：*這 個 房間 的 光線 不 好 ， 不 適合 做 書房 。*

客文：*這 隻 房間 个 光線 毋 好 ， 毋 適合 做 書房 。*

因為我們的目的是要訓練出客語語言模型，因此依照客語斷詞結果，訓練出客語的 Uni-Gram 語言模型，建置方式如表二十五(標點符號不計)：

表二十五：非國客語對應式的客語語言模型範例一

W_i	$C(W_i)$	*Word Probability*
這	1	0.0909090909090909
隻	1	0.0909090909090909
房間	1	0.0909090909090909
个	1	0.0909090909090909
光線	1	0.0909090909090909
毋	2	0.1818181818181818
好	1	0.0909090909090909
適合	1	0.0909090909090909
做	1	0.0909090909090909
書房	1	0.0909090909090909
Totals	11	1

此客語語言模型，即可應用在中文詞轉客文詞的候選詞之序列的選擇上。依照此模型查出的機率值，來找出最好的候選詞之序列，意即找出每個W_i之各別的機率連乘後，機率值為最大的序列。當遇到某個 Event 在此模型中查不到機率時，可用上述的平滑化方法來給予一個機率值，解決零機率之問題。

但是上述模型的訓練方式，若遇到在訓練語料中有多個國客語對應的配對，出現各配對有「中文詞不同但客語詞相同」的情況下，會有「都被當成同一個客語詞」來訓練的問題。如此一來會失去原來國、客語對應的資訊。我們舉個例子來說明(空白處為斷詞邊界)，假設語料中有兩句句子：

中文：*今天 的 風 吹 起來 很 舒服 。*

客文：*今晡日 个 風 吹 起來 當 鬆爽 。*

中文：*這 張 椅子 坐 起來 很 舒適 。*

客文：*這 張 凳仔 坐 起來 當 鬆爽 。*

依照上例的訓練方式，我們針對客文部分做訓練，訓練結果如表二十六的例子：

表二十六：非國客語對應式的客語語言模型範例二

W_i	$C(W_i)$	*Word Probability*
今晡日	1	0.0714285714285714
个	1	0.0714285714285714
風	1	0.0714285714285714
吹	1	0.0714285714285714
起來	2	0.1428571428571429
當	2	0.1428571428571429
鬆爽	2	0.1428571428571429
這	1	0.0714285714285714
張	1	0.0714285714285714
凳仔	1	0.0714285714285714
坐	1	0.0714285714285714
Totals	14	1

由此例可發現，中文詞「舒服」與「舒適」都被當成客語詞「鬆爽」來訓練，這類情況會造成國、客語對應的資訊喪失。因此我們將訓練方式改為「國、客語對應」為單位來訓練，來解決此問題。如表二十七的例子：

表二十七：國客語對應式語言模型範例

W_i	$C(W_i)$	*Word Probability*
今天/今晡日	1	0.0714285714285714
的/个	1	0.0714285714285714
風/風	1	0.0714285714285714
吹/吹	1	0.0714285714285714
起來/起來	2	0.1428571428571429
很/當	2	0.1428571428571429
舒服/鬆爽	1	0.0714285714285714
這/這	1	0.0714285714285714
張/張	1	0.0714285714285714
椅子/凳仔	1	0.0714285714285714
坐/坐	1	0.0714285714285714
舒適/鬆爽	1	0.0714285714285714
Totals	14	1

其中「舒服/鬆爽」與「舒適/鬆爽」因為訓練方式改為國客語對應為一個單位，因此可分辨出兩者是不同的 Event，保留住國客語對應的資訊。

第五章 客語斷詞方法

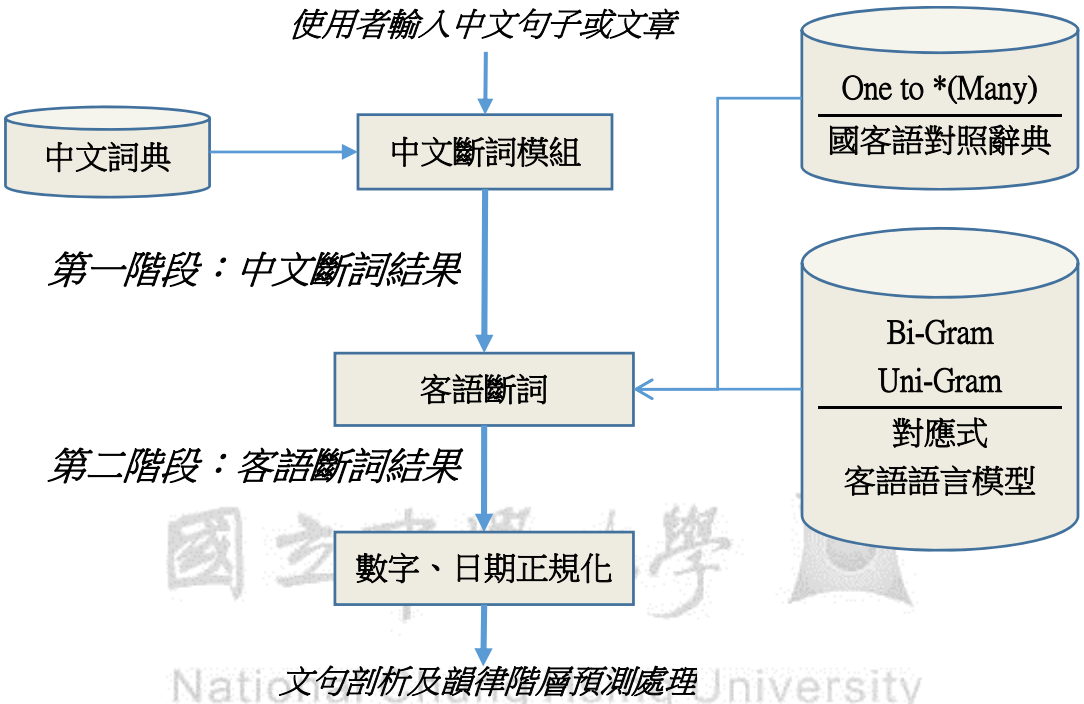
語音合成系統中，斷詞模組是一個相當重要的模組，它將影響著後續的讀音求取及韻律分析的結果。若是輸出的詞性及斷詞邊界錯誤，該文句讀音的求取、文句的剖析及韻律階層的預測，也極有可能造成一連串的錯誤，導致合成出錯誤的語音，甚至沒辦法表現出正確的文意及韻律。因此，一個良好的客語斷詞模組，是提升客語語音合成效果相當重要的部分。在本章中，我們會介紹客語斷詞，以及本論文實作的各項方法和實驗的過程與結果。

5.1 客語斷詞介紹

5.1.1 客語斷詞

目前客語斷詞的做法可分為兩類，第一類為「輸入是客文」，第二類為「輸入是中文」。本論文所提出的客語斷詞方法，是屬於第二類。這類系統容易使用，因為使用者不需要透過客語拼音輸入法來輸入句子，也不需要相當熟悉客語，即可輸入熟悉的中文到系統中，來得到客語斷詞結果。這類的系統應用廣泛，也非常適合讓客語初學者使用。但相較於第一類的系統，實作上會較困難，且斷詞的正確率會較低。因為第二類的系統需要考慮到中文詞轉客語詞的方法，而第一類只是

在客語文句上，找出其斷詞的邊界，因此完全不需要做詞彙翻譯的處理，正確率通常也會較高。圖十七為本論文客語斷詞方法的架構圖(使用語言模型)：



圖十七：客語斷詞方法架構

我們的客語斷詞流程分為兩階段，第一階段是先將中文文句以中文斷詞模組得到斷詞及詞性標記的結果後，再以一部一對多的國客語對照辭典找出所有可能被轉換的客語詞序列，並透過語言模型找出機率或分數最高的客語詞序列，來得到第二階段客語斷詞結果。這個結果可透過文句剖析器處理文句的剖析，來得到更多句法結構上的訊息，這些訊息也是韻律訊息分析相當重要的基礎。因此，一個好的文句分析系統，斷詞模組的效能是相當重要也是最根本的一環。

5.1.2 客語詞性標記

我們的詞性標記方法，是直接採用中文斷詞器所標記上的詞性。本系統所標記的客語斷詞語料，其詞性大多數是使用斷詞器自動標記的，且斷詞器的詞性標記 F 分數已達 92.04%，我們認為該結果相當可靠。因此直接採用斷詞器標記上的詞性，做為客語詞性標記的結果。我們所採用的詞性標記標準，是採用中研院資訊科學所詞庫小組所編列的中研院平衡與料庫(ASBC)詞類標記集的簡化詞類。這個集合所包含的詞性如下表所示：

表二十八：中研院平衡與料庫詞類標記集(簡化詞類)

詞性標籤	詞性意義	詞性標籤	詞性意義
A	非謂形容詞	Neu	數量定詞
Caa	對等連接詞	Nf	量詞
Cab	連接詞	Ng	後置詞
Cba	連接詞	Nh	代名詞
Cbb	關聯連接詞	P	介詞
D	副詞	SHI	是
Da	數量副詞	T	語助詞
DE	的、之、得、地	VA	動作不及物動詞
Dfa	動詞前程度副詞	VAC	動作使動動詞
Dfb	動詞後程度副詞	VB	動作類及物動詞
Di	時態標記	VC	動作及物動詞
Dk	句副詞	VCL	動作接地方賓語動詞
FW	外文標記	VD	雙賓動詞
I	感嘆詞	VE	動作句賓動詞
Na	普通名詞	VF	動作謂賓動詞
Nb	專有名詞	VG	分類動詞

Nc	地方詞	**VH**	狀態不及物動詞
Ncd	位置詞	**VHC**	狀態使動動詞
Nd	時間詞	**VI**	狀態類及物動詞
Nep	指代定詞	**VJ**	狀態及物動詞
Neqa	數量定詞	**VK**	狀態句賓動詞
Neqb	後置數量定詞	**VL**	狀態謂賓動詞
Nes	特指定詞	**V_2**	有

5.2 實驗資源與評估方法

本節將介紹客語斷詞所使用到的訓練(Training Set)及測試(Testing Set)資料集，以及詳細說明我們客語斷詞及詞性標記的評估方法。

5.2.1 實驗資源

本實驗所使用的客語斷詞訓練與測試語料，是採用客委會四縣腔初級及中高級語料中的例句，共 6640 句句子。這份語料分為語料 A 及語料 B，語料 A 有 4196 句，語料 B 有 2444 句，並再將 B 語料依編號順位分為 4 份，B1-B4，每份 611 句。

而這些標記完成的資料，再經過一次經人工篩選後，確認有效句數為：訓練語料 4018 句，測試(B1-B4)語料共 1282 句，我們使用語料的分佈如表二十九所示：

表二十九：客語斷詞語料的使用分佈

	訓練資料集	測試資料集
句數	4018	1282
詞數	45304	17646
字數	65572	25478

5.2.2 評估方法

我們實驗的方法，是「輸入中文未斷詞的句子」到客語斷詞處理演算法中，得到該句的「客語斷詞及標詞性」的輸出，再評估輸出的斷詞結果與正確答案之間對了幾個，並算出正確率。

在斷詞效能評估的方法，我們使用精確率(Precision)、召回率(Recall)、以及 F-分數(F-score)來評估系統的效能，這三種方法的定義如下所示：

$$精確率 = \frac{系統正確斷出的詞數}{系統斷出的總詞數} \quad (5\text{-}2)$$

$$召回率 = \frac{系統正確斷出的詞數}{標準答案的總詞數} \quad (5\text{-}3)$$

$$F - 分數 = \frac{2 \times 精確率 \times 召回率}{精確率 + 召回率} \quad (5\text{-}4)$$

我們以「這個房間的光線不好」為例子(表三十)，來說明如何評估客語斷詞效能(斜線處為斷詞邊界)。

表三十：客語斷詞評估範例

輸入句子	以前人母不夠吃，只能煮番薯飯來充飢。
標準答案	頭擺/人/米/毋/罅/食/，/僅可/煮/蕃薯/飯/來/充飢/。
系統輸出	頭擺/人/米/毋/罅/食/，/僅可/煮/番薯飯/來/充飢/。

系統正確斷出的詞數：12，系統斷出的總詞數：13，標準答案的總詞數：14
精確率：0.923，召回率：0.857， F-分數：0. 889

在詞性標記效能評估的方法，我們同樣也使用精確率(Precision)、召回率(Recall)、以及 F-分數(F-score)來評估系統的效能，這三種方法的定義如下所示：

$$精確率 = \frac{系統正確斷出且詞性標記正確的詞數}{系統斷出的總詞數} \quad (5\text{-}5)$$

$$召回率 = \frac{系統正確斷出且詞性標記正確的詞數}{標準答案的總詞數} \quad (5\text{-}6)$$

$$F - 分數 = \frac{2\times 精確率\times 召回率}{精確率+召回率} \quad (5\text{-}7)$$

我們以「這個房間的光線不好」為例子(表三十一)，來說明如何評估客語斷詞效能。

表三十一：詞性標記評估範例

輸入句子	這個房間的光線不好
標準答案	壁(VC)/柑仔(Na)/愛(D)/對(P)/尾篤項(Nc)/壁(VC)
系統輸出	壁柑仔(VC)/愛(D)/對(P)/尾項(Nc)/壁(VC)
系統正確斷出且詞性標記正確的詞數：3 系統斷出的總詞數：5 標準答案的總詞數：6	
精確率：0.6，召回率：0.5， F-分數：0.545	

除了上述的評估方法外，我們也採用編輯距離演算法(Levenshtein Distance)，來評估轉換後的客語句子與正確答案的客語句子之間的相似程度(Similarity)。此演算法為計算兩字串 A、B 間，由字串 A 轉換成

字串 B 的最小編輯距離(Insertions, Deletions, Substitutions)，計算方式如下：

$$\mathrm{D}(i,j) = \min \begin{cases} D(i-1,j) + InsertCost(Target[i]) \\ D(i-1,j-1) + SubstituteCost(Source[j],\ Target[i]) \\ D(i,j-1) + DeleteCost(Source[j]) \end{cases} \quad (5\text{-}8)$$

其中我們定義每個操作的成本(Cost)為(各種操作的成本可以自訂)：

$$SubstituteCost = \begin{cases} 0 & if\ \ Target[i] = Source[j] \\ 1 & otherwise \end{cases} \quad (5\text{-}9)$$

$$InsertCost = 1 \quad (5\text{-}10)$$

$$DeleteCost = 1 \quad (5\text{-}11)$$

演算法執行完得到最後的最小編輯距離後，由公式(5-12)將距離轉換成 0 到 1 之間的值，即為 A、B 字串間的相似度。其中$Max\ Length(A,B)$為取 A、B 兩字串之最長長度的字數：

$$Similarity(A,B) = 1 - \frac{D(A,B)}{Max\ Length(A,B)} \quad (5\text{-}12)$$

我們採用這個評估方法有兩個原因：

原因一：

中文翻客文的處理，是一個中文詞對多種可能客語字詞的問題，且客語常有一意多詞的問題，如：中文「收到」客語可翻成「收」或「收着」。若正確答案標記為「收着」，但系統輸出為「收」，兩者詞不同，但意思卻相同。如此一來，使用第一種的述評估方法，其正確率為 0%，但以相似程度的角度來看，兩者僅差一個字，相似度仍有 50%。因此除了斷詞的評估標準外，其字串轉換後的相似度也可以是一個評估效能的標準。

原因二：

本論文的客語斷詞模組，使用於客語語音合成系統中，因此翻譯後的字串，距離標準答案的相似度越高，能將字唸對的可能性也將越高。

我們以「這個房間的光線不好」為例，來說明相似度的評估方法。此評估法我們只評估斷詞結果，並未評估詞性標記結果。評估的方式是將正確答案與系統輸出答案的斷詞邊界拿掉，變成連續的字串，如表三十二所示。

表三十二：客語斷詞相似度評估範例

正確答案	這隻房間个光線毋好
系統答案	這個間房个光線毋好
編輯距離	3
字串相似度	$1 - 3/9 = 0.666$

這個例子中，客語斷詞的正確答案的字串為「這隻房間个光線毋好」，但系統輸出的字串為「這個間房个光線毋好」，透過公式(5-8)來計算其編輯距離，詳細的計算結果如表三十三所示：

表三十三：客語斷詞相似度評估計算範例

	Target	這	隻	房	間	个	光	線	毋	好
Source	0	1	2	3	4	5	6	7	8	9
這	1	0	1	2	3	4	5	6	7	8
個	2	1	1	2	3	4	5	6	7	8
間	3	2	2	2	2	3	4	5	6	7
房	4	3	3	2	3	3	4	5	6	7
个	5	4	4	3	3	3	4	5	6	7
光	6	5	5	4	4	4	3	4	5	6
線	7	6	6	5	5	5	4	3	4	5
毋	8	7	7	6	6	6	5	4	3	4
好	9	8	8	7	7	7	6	5	4	***3***

在上表中，此演算法的走訪過程為：由上而下，由左而右。若$Source[j]$等於$Target[i]$，也就是兩字元相等，則它們的 Cost 為 0，否則為 1，如公式(5-9)。而每格$D(i,j)$的最短編輯距離，其計算方式是本身的 Cost(即$Source[j]$是否等於$Target[i]$之判斷後所得到的 Cost 加已累積的 Cost)加上該字元所在位置的$D(i-1,j)$、$D(i-1,j-1)$及$D(i,j-1)$三者的最小值，如公式(5-8)。最後我們得到兩字串間的編輯距離為 3，再透過公式(5-12)計算出其相似度為 66.6%。

5.3 修改中文斷詞辭典

本論文所提出之方法，斷詞邊界在第一階段就已被決定，因此為了讓一些客語用詞有機會成為被轉換的候選詞，我們將國客語對照辭典中，所有詞長大於 1 的中文詞，都加入到中文斷詞辭典裡。如此做法能提高原本完全不會被找到的客語詞,有機會成為被轉換的候選詞。在此我們列舉兩個例子說明 (斜線處代表斷詞邊界)：

表三十四：中文斷詞邊界的限制範例一

中文句子	泥鰍滑溜溜
中文斷詞	泥鰍/***滑***/***溜溜***
系統答案	鰗鰍仔/***滑***/***溜溜***
正確答案	鰗鰍仔/***滑溜溜仔***

如表三十四的範例一所示，透過中文斷詞得到「泥鰍/滑/溜溜」的斷詞邊界，再透過客語斷詞處理方法，將中文斷詞結果輸出為「鰗鰍仔/滑/溜溜/」。但其實我們的國客語對照詞典中，有收錄「滑溜溜/滑溜溜仔」這筆國客語對照資料，礙於中文斷詞邊界的限制，只能由原本的一個詞「滑溜溜仔」轉成「滑/溜溜」兩個詞。此情況會直接的影響到轉換的正確率。

表三十五：中文斷詞邊界限制範例二

中文句子	大家來去客家庄走一走。
中文斷詞	大家/來去/客家庄/***走/一/走***
系統答案	大家/來去/客家庄/***行/一/行***
正確答案	大家/來去/客家庄/***遶遶啊***

如表三十五的範例二所示，「走一走」被斷成「走/一/走」，但實際上我們詞典有收錄「走一走/遶遶啊」這筆國客語對應資料，礙於中文斷詞邊界，我們沒辦法選到該筆對應。

基於以上原因，我們決定將國客語對照辭典中的中文詞，加入到中文斷詞辭典中。而這些新加入的中文詞，要決定出它們的詞頻大小，因為這兩個語料規模相差非常龐大，我們的中文斷詞辭典總詞頻數高達 462729801 個詞，而客語訓練語料僅有 45304 個詞。因此我們將中文辭典的平均詞頻取Log_2乘上 15，再與國客語對照辭典的詞頻相乘，得到該詞新的詞頻。步驟如下：

步驟一：統計出中文斷詞辭典原始的分佈：

表三十六：中文斷詞辭典詞頻分佈

總詞數	995642
總詞頻	462729801
平均詞頻	464

步驟二：計算出每個要加入中文斷詞辭典中的中文詞，其新詞頻 $\mathbf{C}^*(\boldsymbol{W_i})$，其中$\boldsymbol{W_i} \in (\boldsymbol{Word\ Length} > \mathbf{1})$：

$$C^*(W_i) = ceil\{log_2(464) \times 15 \times [C(W_i) + 1]\} \quad (5\text{-}1)$$

表三十七為一個國客對照詞「走一走/遶遶啊」，其詞頻轉換的例子：

表三十七：國客語對照辭典的中文詞詞頻換算

中文詞	走一走
客語詞	遶遶啊
$\boldsymbol{C(W_i)}$	1
$\boldsymbol{C^*(W_i)}$	$Ceil[log_2(464) \times 15 \times (1+1)] = 134$

依照上述方法，產生以下新增候選列表，如表三十八：

表三十八：客語詞新詞頻候選表

中文詞	**詞頻**
走一走	134
滑溜溜	134
…	…

步驟三：將步驟二產生的結果加入中文斷詞辭典，若有相同的中文詞則其詞頻相加。

最後我們評估修改前與修改後的中文斷詞系統，其正確率的差異。測試為外部測試，使用「中研院平衡語料庫 3.0 (Academia Sinica Balanced Corpus 3.0 Version)」。如表三十九：

表三十九：中文斷詞辭典修改前後的斷詞正確率比較

	Precision	Recall	F-Measure
修改前	97.16%	96.21%	96.69%
修改後	97.15%	96.16%	96.65%

可發現 F-Measure 略低 0.04%，但改善了以下問題：

表四十：加入客語詞後的中文斷詞辭典的改善

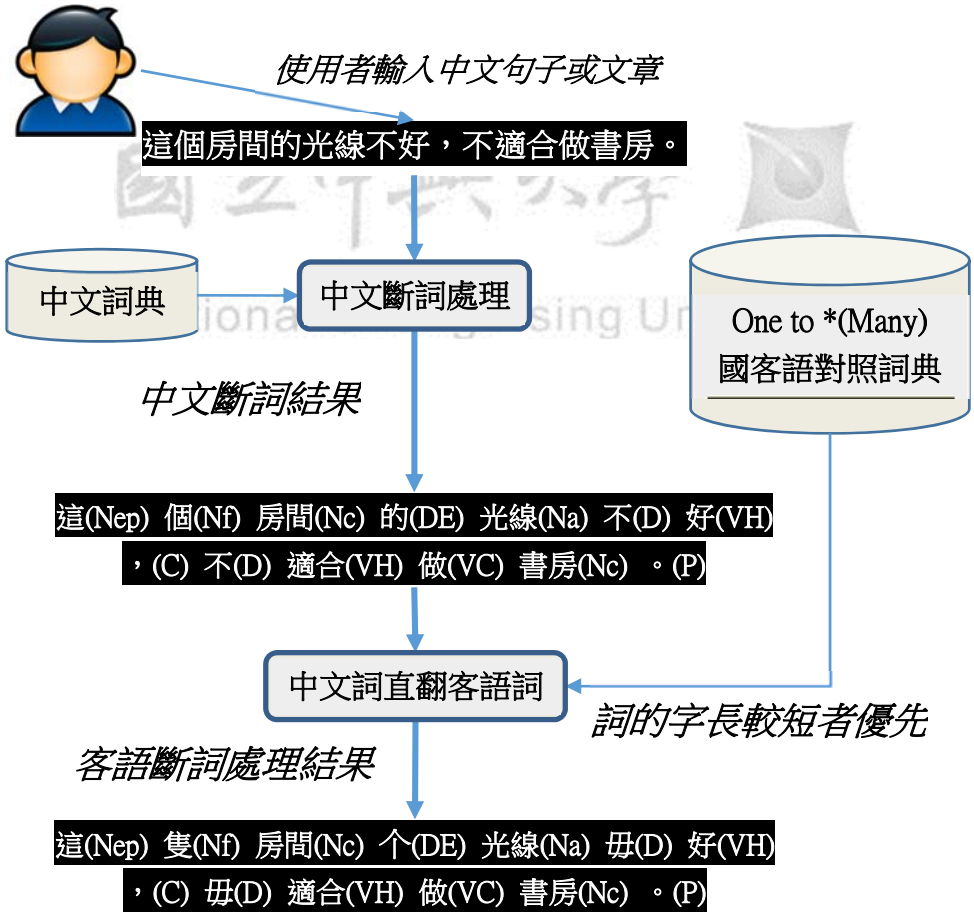
中文句子	泥鰍滑溜溜，
原中文斷詞	泥鰍/滑/溜溜/，
改善後中文斷詞	泥鰍/滑溜溜/，
原中文翻客語	鰗鰍仔/滑/溜溜/，
改善後中文翻客語	鰗鰍仔/滑溜溜仔/，
正確答案	鰗鰍仔/滑溜溜仔/，

原本的斷詞系統無法將「滑溜溜」判斷出來，修正辭典後已能正確斷出，並轉為客語詞「滑溜溜仔」。

5.4 中文詞直翻客語詞

5.4.1 斷詞方法敘述

此方法未使用任何語言模型，僅透過中文斷詞模組得到第一階段的斷詞邊界後，再透過國客語對照辭典，以及詞的字長較短詞優先的原則，將中文詞轉換成客語詞。當中文詞找不到對應的客語詞時，則直接使用原中文詞。方法如圖十八所示：



圖十八：中文直翻客語詞之斷詞方法示意圖

5.4.2 實驗結果與討論

這項實驗主要目的，是觀察國客語對照辭典詞目的增加及校正，是否會影響中文詞轉客語詞的效能，我們使用不同的辭典進行下列三項實驗：

實驗 A：使用羅丞邑[34]的國客語對照辭典(27514 個詞目)。

實驗 B：使用吳俊毅[15]的國客語對照辭典(26993 個詞目)。

實驗 C：使用本論文所建置的國客語對照詞典(42772 個詞目)。

表四十一：中文詞直翻客語詞斷詞方法實驗結果

	Precision	Recall	F- Measure	字串相似度
實驗 A	68.41%	69.12%	68.76%	73.29%
實驗 B	72.66%	73.42%	73.04%	75.68%
實驗 C	75.02%	75.80%	75.41%	78.36%

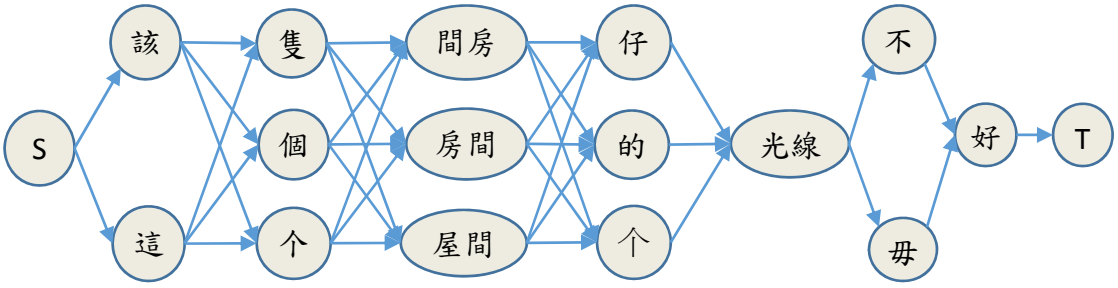
由此實驗結果可得知，詞典的校正與詞目的增加，確實能顯著提升斷詞的效能。

5.5 使用動態規劃法及語言模型的客語斷詞

因為用來訓練語言模型的語料相當稀少，存在嚴重資料稀疏的問題。因此，若要使用客語語言模型，我們必須找出最適當的平滑化方法，以發揮現有極小量語料的最大效益。

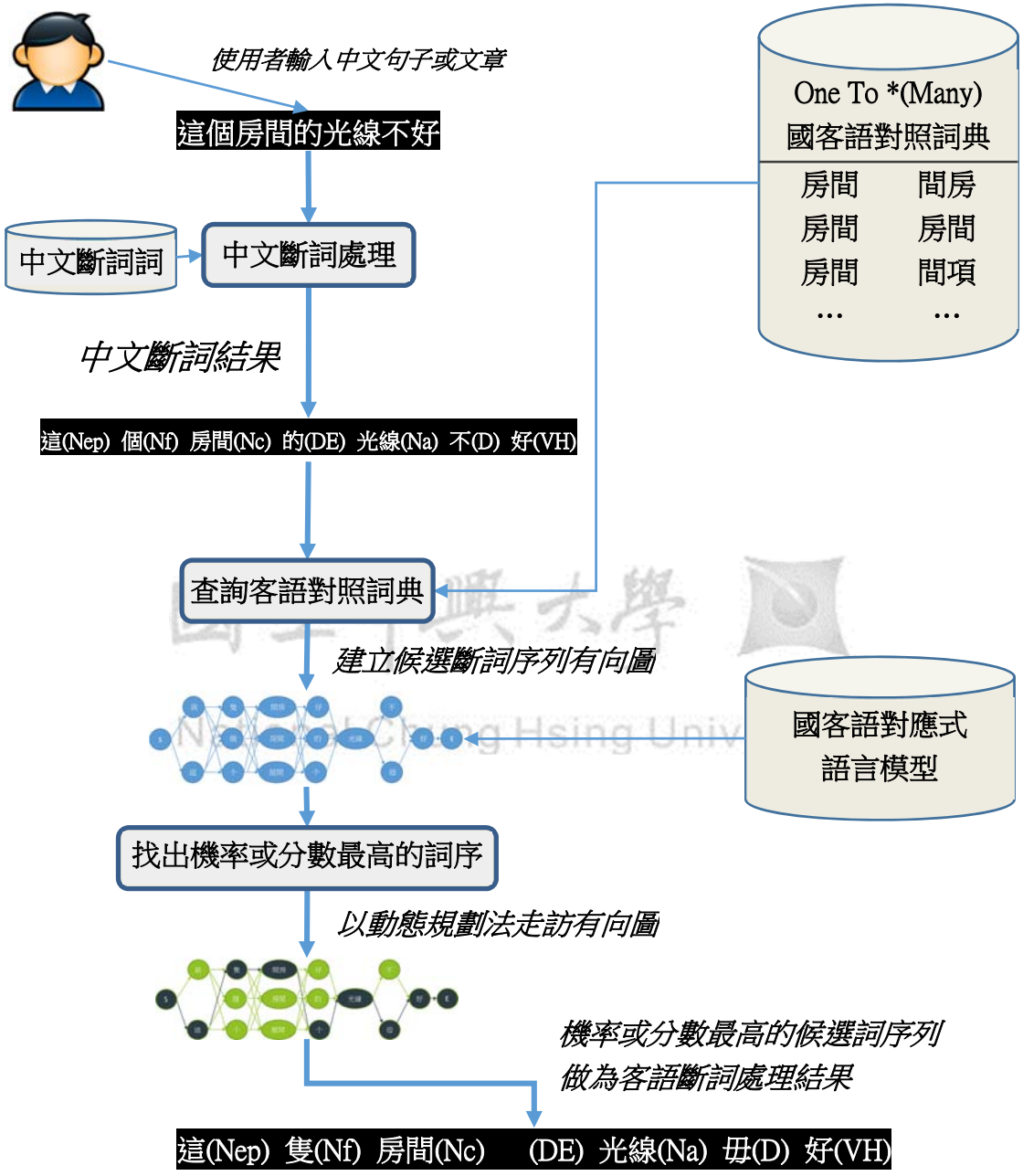
5.5.1 斷詞方法敘述

此方法是先透過中文斷詞模組得到第一階段的中文斷詞及詞性標記結果，再以一對多的國客語對照辭典，找出每個中文詞所有可能被轉換的客語詞、建立出斷詞序列有向圖，再以各種不同平滑法處理過的國客語對應式語言模型，來找出所有可能詞串$\widehat{W} = W_1, W_2, \ldots, W_n$中，擁有最大機率$arg\,max\ P(W_1, W_2, W_3, \ldots, W_n)$的詞序列。圖十九是中文短句：「這個房間的光線不好」，經中文斷詞處理及從國客語對照辭典，找出所有可能被轉換的客語候選詞後，所建立出的有向圖。該圖可表示該句多個可能客語詞之間的前後關係，我們要透過語言模型及動態規劃演算法，來找出機率最佳的詞序列。



圖十九：句子中可能詞之間的前後關係圖

接下來我們將詳細說明應用各種平滑法的客語斷詞，其做法與實驗結果。圖二十為使用語言模型的客語斷詞架構圖。



圖二十：使用語言模型的客語斷詞示意圖

詞性標記方法，是直接採用中文斷詞模組的詞性標記結果。因為其正確率達 92.04%，錯誤率相當低，我們認為此結果已相當可靠。

5.6 使用 Uni-Gram 加成平滑法的客語斷詞

本斷詞法為使用動態規劃及語言模型的斷詞方法，這節主要探討的是找出最佳的平滑模型及實際測試其效果。

5.6.1 找出最佳的δ值

在加成平滑法中，我們必須找出一個最佳的$\delta\ (0 < \delta \leq 1)$值，$\delta$值越小，代表已知事件分出給未知事件的機率量越少。但我們並不知道要分出多少的機率量，才是最佳的平滑模型。因此我們透過交叉熵找出能產生最低混淆度的模型。我們的實驗方法為使用訓練語料產生$0.1 \leq \delta \leq 1.0$的 Uni-Gram 加成平滑法模型，並利用測試語料來觀察該模型在$0.1 \leq \delta \leq 1.0$之間的混淆度，如表四十二。

表四十二：Uni-Gram 加成平滑法模型的外部測試混淆度

$\boldsymbol{\delta}$	$\boldsymbol{H(T)}$	$\boldsymbol{PP(T)}$
1.0	11.1859012906901	2329.65610597304
0.9	11.155033200094	2280.33994996924
0.8	11.1243635205926	2232.3747537267
0.7	11.0944706556933	2186.59551703388
0.6	11.0663110368262	2144.32971291395
0.5	11.0415295495254	2107.81063982828
0.4	11.0231341351154	2081.10513539101
0.3	***11.0171843947382***	***2072.54023572396***
0.2	11.0380690381436	2102.76081189035
0.1	11.1335486669436	2246.63286105947

我們觀察到當$\delta = 0.3$時，有最佳的混淆度，因此採用此模型來進

行接下來客語斷詞的實驗。

5.6.2 實驗結果與討論

我們將$\delta = 0.3$的 Uni-Gram 平滑模型，實際應用於客語斷詞。並分別做了斷詞及詞性標記的內外部測試，表四十三及四十四分別為內部測試及外部測試的實驗結果。

表四十三：Uni-Gram 加成平滑模型內部測試結果

內部測試	Precision	Recall	F-Measure	相似度
斷詞	87.82%	87.14%	87.48%	90.72%
詞性標記	87.14%	86.46%	86.80	-

表四十四：Uni-Gram 加成平滑模型外部測試結果

外部測試	Precision	Recall	F-Measure	相似度
斷詞	81.11%	79.85%	80.47%	84.28%
詞性標記	80.70%	79.44%	80.06%	-

由本實驗顯示，相較於未使用語言模型的傳統方法，使用語言模型可顯著提升客語斷詞的效能。

5.7 使用 Uni-Gram 凱氏平滑法的客語斷詞

這節主要探討的是利用凱氏 Uni-Gram 平滑模型來實際測試其效果，並與 5.6 節的加成平滑模型做比較。

5.7.1 找出 Cut-off k 值

在凱氏平滑法中，我們必須找出 Cut-off 的 k 值，意即訓練語料中出現次數為$1 \le c \le k$的 events，都必須透過一個折扣函數$d'(c)$，折扣出機率給未知事件，以解決零機率的問題。當$c > k$時，則不需要分出任何機率。

這個 k 值可透過 Good-Turing Estimation 公式(4-15)觀察得到。在觀察 k 值的過程中，凡經過調整後的c_{GT}^{*}，必須符合下列兩項性質。若不符合，則前一項的c即為 Cut-off 的終止值 k：

1. 調整後的c_{GT}^{*}不得大於原來的次數c。
2. 調整後的c_{GT}^{*}與原來次數c的比例必須比前項高。

以下為我們實際使用 Uni-Gram 模型，找出 cut-off k 值的過程：

我們用來訓練 Uni-Gram 模型的語料有 45284 個 Token、10477 個 Uni-Gram 事件，客語辭典有 42772 個詞，因此出現次數為 0 次的事件 (Unseen Events)，有$n_0 = 42772 - 10477 = 32295$個。透過統計出 Uni-Gram 各事件出現的次數分佈n_c，以及使用公式(4-15)來計算出各

次數的c^*_{GT}，可得到下列表格。並觀察出在$c = 5$時，c^*_{GT}/c的比例出現遞減情形，因此前一項$c = 4$為 Cut-off 的終止，$k = 4$。結果如表四十五。

表四十五：Uni-Gram 語言模型的分佈($0 \le c \le 9$)

原始次數，c	調整後的次數，c^*_{GT}	事件數，n_c	調整後比例，c^*_{GT}/c
0	0.215915776435981	32295	-
1	0.442850996701563	6973	0.442850996701563
2	1.20660621761658	1544	0.60330310880829
3	2.07407407407407	621	0.691358024691358
4	***3.1055900621118***	***322***	***0.77639751552795***
5	3.75	200	0.75
6	5.32	125	0.886666666666667
7	6.14736842105263	95	0.878195488721804
8	6.28767123287671	73	0.785958904109589
9	7.45098039215686	51	0.827886710239651

我們找出 k 值後，即可求出在$1 \le c \le 4$之間，其折扣函數$d'(c)$的折扣值，如表四十五：

表四十六：$k = 4$的凱氏 Uni-Gram 模型之n_c分佈折扣函數

c	1	2	3	4	5
n_c	6973	1544	621	322	200
c^*_{katz}	0.3495	1.0737	1.9190	2.9558	-
$d'(c)$	0.3495	0.5369	0.6396	0.7390	-

詳細的計算過程如下：

$$1^*_{katz} = \frac{(1+1)\times\frac{n_{1+1}}{n_1} - 1\times\frac{(4+1)\times n_{4+1}}{n_1}}{1-\frac{(4+1)\times n_{4+1}}{n_1}} = \frac{2\times\frac{1544}{6973} - \frac{5\times 200}{6973}}{1-\frac{5\times 200}{6973}} = 0.3495$$

$$d'(1) = \frac{c^*_{katz}}{c} = \frac{0.3495}{1} = 0.3495$$

$$2^*_{katz} = \frac{(2+1)\times\frac{n_{2+1}}{n_2} - 2\times\frac{(4+1)\times n_{4+1}}{n_1}}{1-\frac{(4+1)\times n_{4+1}}{n_1}} = \frac{3\times\frac{621}{1544} - 2\times\frac{5\times 200}{6973}}{1-\frac{5\times 200}{6973}} = 1.0737$$

$$d'(2) = \frac{c^*_{katz}}{c} = \frac{1.0737}{2} = 0.5369$$

$$3^*_{katz} = \frac{(3+1)\times\frac{n_{3+1}}{n_3} - 3\times\frac{(4+1)\times n_{4+1}}{n_1}}{1-\frac{(4+1)\times n_{4+1}}{n_1}} = \frac{4\times\frac{322}{621} - 3\times\frac{5\times 200}{6973}}{1-\frac{5\times 200}{6973}} = 1.9190$$

$$d'(3) = \frac{c^*_{katz}}{c} = \frac{1.9190}{3} = 0.6396$$

$$4^*_{katz} = \frac{(4+1)\times\frac{n_{4+1}}{n_4} - 4\times\frac{(4+1)\times n_{4+1}}{n_1}}{1-\frac{(4+1)\times n_{4+1}}{n_1}} = \frac{5\times\frac{200}{322} - 4\times\frac{5\times 200}{6973}}{1-\frac{5\times 200}{6973}} = 2.9558$$

$$d'(4) = \frac{c^*_{katz}}{c} = \frac{2.9558}{4} = 0.7390$$

得到每個出現次數為c的折扣值後，我們即可算出這些事件所折扣出來的機率量，如下表四十七所示：

表四十七：$k = 4$的凱氏 Uni-Gram 語言模型$(1 \le c \le 4)$

C	原機率值	折扣後機率值	事件數，n_c	分出機率量
1	2.2082×10^{-5}	7.7195×10^{-6}	6973	0.10015
2	4.4165×10^{-5}	2.3710×10^{-5}	1544	0.031583
3	6.6248×10^{-5}	4.2378×10^{-5}	621	0.014823
4	8.8331×10^{-5}	6.5272×10^{-5}	322	0.007425
總計分出的機率量				***0.153981***

詳細的計算過程如下：

$$P_c^* = \frac{c_{katz}^*}{N} = \frac{c \times d'(c)}{N}$$

$N = 45284$，為 Uni-Gram 模型所有的 Tokens。

$$P_1^* = \frac{1 \times d'(1)}{N} = \frac{1 \times 0.3495}{45284} = 7.7195 \times 10^{-6}$$

$$P_2^* = \frac{2 \times d'(2)}{N} = \frac{2 \times 0.5369}{45284} = 2.3710 \times 10^{-5}$$

$$P_3^* = \frac{3 \times d'(3)}{N} = \frac{3 \times 0.6396}{45284} = 4.2378 \times 10^{-5}$$

$$P_4^* = \frac{4 \times d'(4)}{N} = \frac{4 \times 0.7390}{45284} = 6.5272 \times 10^{-5}$$

最後，我們可求出次數為 0 次的事件，它們所分到的總機率量：

$$P_{GT}^* = P_0^* \times n_0 = (0+1) \times \frac{n_1}{n_0} \times \frac{1}{N} \times n_0 = \frac{n_1}{N} = \frac{6973}{45284} = 0.153981$$

而我們 Uni-Gram 模型中的未知事件(Unseen Events)有 32295 個，因此每個未知事件所分得的機率量為：

$$P_0^* = \frac{P_{GT}^*}{N} = \frac{0.153981}{32295} = 4.7680 \times 10^{-6}$$

5.7.2 實驗結果與討論

我們將$k = 4$的凱氏 Uni-Gram 模型，實際應用於客語斷詞處理。並分別做了斷詞及詞性標記的內外部測試，表四十八及四十九分別為內部測試及外部測試的實驗結果。

表四十八：Uni-Gram 凱氏平滑模型內部測試結果

內部測試	Precision	Recall	F-Measure	相似度
斷詞	87.82%	87.14%	87.48%	90.72%
詞性標記	87.14%	86.46%	86.80%	-

表四十九：Uni-Gram 凱氏平滑模型外部測試結果

外部測試	Precision	Recall	F-Measure	相似度
斷詞	81.11%	79.85%	80.47%	84.28%
詞性標記	80.70%	79.44%	80.06%	-

實驗結果顯示，在 Uni-Gram 語言模型中，使用加成平滑法和凱氏平滑法的 F 分數一樣。我們實際檢視兩個方法的斷詞結果，也未出現同正確率但不同斷詞結果的情況，證實斷詞結果完全相同。

造成此結果的原因，是因為我們的語言模型相當稀疏，即便理論上凱氏平滑法的平滑效果比加成平滑法更 Smoothing。但平滑後的機率模型，因樣本數太少而造成其機率分佈情況相同，因此用在斷詞候選詞序列的選擇上，效果也相同。

5.8 使用 Bi-Gram 強化凱氏平滑法的客語斷詞

關於 Bi-Gram 語言模型，我們不採用加成平滑法來做平滑。因為目前客語語料非常稀疏，在 Bi-Gram 中更是如此。因此，若使用沒有 Back-off 架構的平滑方法，會造成當事件出現次數為零時，也就是當$C(W_{i-1}, W_i) = 0$時，其未知事件的機率$P_{add}(W_i|W_{i-1})$都一樣，無法依照$C(W_i)$的分佈，來分辨出每個事件的重要性。表五十是一個詞典詞數$V = 15$(舉例)、$\delta = 1$，在對應式客語語言模型中，$C(房間/間房, W_k)$所有事件的計算範例。依照公式(5-14、5-15)計算：

表五十：使用加成平滑法的 Bi-Gram 模型範例

W_{i-1}	W_i	$C(W_{i-1}, W_i)$	$C(W_i)$	$P_{add}(W_i\|W_{i-1})$
房間/間房	，/，	2		1.25×10^{-1}
房間/間房	。/。	1		8.33×10^{-2}
房間/間房	光線/光線	1		8.33×10^{-2}
房間/間房	有/有	1		8.33×10^{-2}
房間/間房	西曬/向西	1		8.33×10^{-2}
房間/間房	那麼/恁	1		8.33×10^{-2}
房間/間房	都/就	1		8.33×10^{-2}
房間/間房	很/當	1		8.33×10^{-2}
房間/間房	前/前	0	28	4.16×10^{-2}
房間/間房	客人/人客	0	12	4.16×10^{-2}
房間/間房	客廳/廳下	0	5	4.16×10^{-2}
房間/間房	建築/建築	0	2	4.16×10^{-2}
房間/間房	漂亮/靚	0	0	4.16×10^{-2}
房間/間房	租賃/租賃	0	0	4.16×10^{-2}
房間/間房	乾淨/乾淨	0	0	4.16×10^{-2}
總合		9	47	1

$$C^*(W_{i-1},W_i)=\frac{\delta+C(W_{i-1},W_i)}{\delta\times V+\sum_{W_k}C(W_{i-1},W_k)}\times\sum_{W_k}C(W_{i-1},W_k) \quad (5\text{-}14)$$

$$P_{add}(W_i|W_{i-1})=\frac{C^*(W_{i-1},W_i)}{\sum_{W_k}C(W_{i-1},W_k)} \quad (5\text{-}15)$$

由表可觀察到，當$C(W_{i-1},W_i)=0$時，不管$C(W_i)$多少，其機率皆為4.16×10^{-2}，無法依照$C(W_i)$來分辨出W_i在語料中的重要性。因此我們僅採用有 Back-off 架構的強化凱氏平滑法，來進行 Bi-Gram 模型的平滑處理。

5.8.1 找出 Cut-off k 值

在 Uni-Gram 的凱氏平滑法中已說明，此方法必須找出 Cut-off 的 k 值，來針對訓練語料中出現次數為$1\le c\le k$的事件進行機率的折扣，並分給未知事件，以解決零機率的問題。我們可透過與 Uni-Gram 相同的方法，找出 k 值。而強化凱氏平滑法是以凱氏平滑法為基礎的方法，強化版只差在有針對當$C(W_i)=0$的情況做平滑處理，可參考公式(4-22)，其於計算方式皆相同，找出 k 值的方式也一樣。以下為我們實際使用 Bi-Gram 模型，找出 k 值的過程：

我們用來訓練 Bi-Gram 模型的語料有 45284 個 Token、29524 個 Bi-Gram 事件，客語辭典有 42772 個詞。透過 Good-Turing Estimation 公式(4-15)，我們觀察出在$c=5$時，c^*_{GT}/c的比例出現遞減情形，因此

找到$k = 4$。表五十一為所示：

表五十一：Bi-Gram 語言模型的分佈($1 \le c \le 5$)

原始次數，c	調整後的次數，c^*_{GT}	詞數，n_c	調整後比例，c^*_{GT}/c
1	0.1932	25379	0.1932
2	0.8576	2452	0.4288
3	1.7917	701	0.5972
4	***3.1847***	***314***	***0.7961***
5	3.33	200	0.666

我們找出 k 值後，可求出在$1 \le c \le 4$之間，其折扣函數$d(c)$的折扣值，如表五十二所示：

表五十二：$k = 4$的凱氏 Bi-Gram 模型之n_c分佈折扣函數

c	1	2	3	4	5
n_c	25379	2452	701	314	200
c^*_{katz}	0.1601	0.8108	1.7421	3.1512	-
$d(c)$	0.1601	0.4054	0.5807	0.7878	-

詳細的計算過程如下：

$$1^*_{katz} = \frac{(1+1)\times\frac{n_{1+1}}{n_1} - 1\times\frac{(4+1)\times n_{4+1}}{n_1}}{1-\frac{(4+1)\times n_{4+1}}{n_1}} = \frac{2\times\frac{2452}{25379} - \frac{5\times 200}{25379}}{1-\frac{5\times 200}{25379}} = 0.1601$$

$$d(1) = \frac{c^*_{katz}}{c} = \frac{0.1601}{1} = 0.1601$$

$$2^*_{katz} = \frac{(2+1)\times\frac{n_{2+1}}{n_2} - 2\times\frac{(4+1)\times n_{4+1}}{n_1}}{1-\frac{(4+1)\times n_{4+1}}{n_1}} = \frac{3\times\frac{701}{2452} - 2\times\frac{5\times 200}{25379}}{1-\frac{5\times 200}{25379}} = 0.8108$$

$$d(2) = \frac{c^*_{katz}}{c} = \frac{0.8108}{2} = 0.4054$$

$$3^*_{katz} = \frac{(3+1)\times\frac{n_{3+1}}{n_3} - 3\times\frac{(4+1)\times n_{4+1}}{n_1}}{1-\frac{(4+1)\times n_{4+1}}{n_1}} = \frac{4\times\frac{314}{701} - 3\times\frac{5\times 200}{25379}}{1-\frac{5\times 200}{25379}} = 1.7421$$

$$d(3) = \frac{c^*_{katz}}{c} = \frac{1.7421}{3} = 0.5807$$

$$4^*_{katz} = \frac{(4+1)\times\frac{n_{4+1}}{n_4} - 4\times\frac{(4+1)\times n_{4+1}}{n_1}}{1-\frac{(4+1)\times n_{4+1}}{n_1}} = \frac{5\times\frac{200}{314} - 4\times\frac{5\times 200}{25379}}{1-\frac{5\times 200}{25379}} = 3.1512$$

$$d(4) = \frac{c^*_{katz}}{c} = \frac{3.1512}{4} = 0.7878$$

表五十三為一個 Bi-Gram 語言模型中，所有W_{i-1}為「晚上/暗晡」的已知事件加上六個未知事件的例子，來說明強化凱氏平滑法的計算方式。

表五十三：使用強化凱氏平滑法的 Bi-Gram 模型範例

W_{i-1}	W_i	$C(W_{i-1}, W_i)$	$C(W_i)$	$P_{EKatz}(W_i \mid W_{i-1})$
晚上/暗晡	才/正	100		0.440528634
晚上/暗晡	失眠/反躁	50		0.220264317
晚上/暗晡	生/降	25		0.110132159
晚上/暗晡	吃飽飯/食飽夜	8		0.035242291
晚上/暗晡	同學會/同學會	9		0.039647577
晚上/暗晡	要/愛	20		0.088105727
晚上/暗晡	做/發	5		0.022026432
晚上/暗晡	都/都	4		0.013881938
晚上/暗晡	給/分	3		0.007674449
晚上/暗晡	跟/跲	2		0.003571806
晚上/暗晡	睡/睡	1		0.000705286
晚上/暗晡	人家/人家	0	4	0.005385536
晚上/暗晡	十分/十分	0	3	0.003495862
晚上/暗晡	上街/上街	0	2	0.001956356
晚上/暗晡	喝酒/食酒	0	1	0.000636754
晚上/暗晡	電影/電影	0	0	0.0033725
晚上/暗晡	工作/工作	0	0	0.0033725
合計		227	10	1

每個事件詳細的計算過程如下：

以下皆為$C(W_{i-1}, W_i) > k$的事件不需要折扣：

$$\mathrm{P}(\textit{才/正} \mid \textit{晚上/暗晡}) = \frac{C(W_{i-1}, W_i)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{100}{227} = 0.440528634$$

$$\mathrm{P}(\textit{失眠/反躁} \mid \textit{晚上/暗晡}) = \frac{C(W_{i-1}, W_i)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{50}{227} = 0.220264317$$

$$P(\text{生/降}|\text{晚上/暗哨}) = \frac{C(W_{i-1}, W_i)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{25}{227} = 0.110132159$$

$$P(\text{吃飽飯/食飽夜}|\text{晚上/暗哨}) = \frac{C(W_{i-1}, W_i)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{8}{227} = 0.035242291$$

$$P(\text{同學會/同學會}|\text{晚上/暗哨}) = \frac{C(W_{i-1}, W_i)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{9}{227} = 0.039647577$$

$$P(\text{要/愛}|\text{晚上/暗哨}) = \frac{C(W_{i-1}, W_i)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{20}{227} = 0.088105727$$

$$P(\text{做/發}|\text{晚上/暗哨}) = \frac{C(W_{i-1}, W_i)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{5}{227} = 0.022026432$$

以下皆為$0 < C(W_{i-1}, W_i) \leq k$的事件，須依照 Bi-Gram 模型的折扣函數$d(c)$折扣出機率：

$$P(\text{都/都}|\text{晚上/暗哨}) = \frac{C(W_{i-1}, W_i) \times d(4)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{4 \times 0.7878}{227} = 0.013881938$$

$$P(\text{給/分}|\text{晚上/暗哨}) = \frac{C(W_{i-1}, W_i) \times d(3)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{3 \times 0.5807}{227} = 0.007674449$$

$$P(\text{跟/趵}|\text{晚上/暗哨}) = \frac{C(W_{i-1}, W_i) \times d(2)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{2 \times 0.4054}{227} = 0.003571806$$

$$P(\text{睡/睡}|\text{晚上/暗哨}) = \frac{C(W_{i-1}, W_i) \times d(1)}{\sum_{W_k} C(W_{i-1}, W_k)} = \frac{1 \times 0.1601}{227} = 0.000705286$$

以下皆為$C(W_{i-1}, W_i) = 0$ and $0 < C(W_i) \leq k$的事件，須透過 Back-off 後的分佈，來分享上列事件總共所分出來的機率量$\alpha(W_{i-1})$。

$$\alpha(W_{i-1}) = 1 - 0.981780617 = 0.018219383$$

因為同時也是$0 < C(W_i) \le k$的事件，須依照 Uni-Gram 模型的折扣函數 $d'(c)$折扣出機率：

$$P(\text{人家/人家}|\textit{晚上/暗晡}) = \alpha(W_{i-1}) \times \frac{C(W_i) \times d'(4)}{\sum_{W_j:C(W_{i-1},W_j)=0} C(W_j)}$$

$$= 0.018219383 \times \frac{4 \times 0.7390}{10} = 0.005385536$$

$$\mathrm{P}(\text{十分/十分}|\textit{晚上/暗晡}) = \alpha(W_{i-1}) \times \frac{C(W_i) \times d'(3)}{\sum_{W_j:C(W_{i-1},W_j)=0} C(W_j)}$$

$$= 0.018219383 \times \frac{3 \times 0.6396}{10} = 0.003495862$$

$$\mathrm{P}(\text{上街/上街}|\textit{晚上/暗晡}) = \alpha(W_{i-1}) \times \frac{C(W_i) \times d'(2)}{\sum_{W_j:C(W_{i-1},W_j)=0} C(W_j)}$$

$$= 0.018219383 \times \frac{2 \times 0.5369}{10} = 0.001956356$$

$$\mathrm{P}(\text{喝酒/食酒}|\textit{晚上/暗晡}) = \alpha(W_{i-1}) \times \frac{C(W_i) \times d'(1)}{\sum_{W_j:C(W_{i-1},W_j)=0} C(W_j)}$$

$$= 0.018219383 \times \frac{1 \times 0.3495}{10} = 0.000636754$$

以下皆為$C(W_{i-1}, W_i) = 0$ and $C(W_i) = 0$的事件，須依照$C(W_i) = 0$的個數 T，來平分 Back-off 的事件所分出的機率量$\beta(W_{i-1})$。

$$\beta(W_{i-1}) = \alpha(W_{i-1}) - \sum_{W_j:C(W_{i-1},W_j)=0 \text{ and } C(W_j)>0} P_{katz}(W_j|W_{i-1})$$

$$= 0.018219383 - 0.011474508 = 0.006744875$$

$$P(\text{電影/電影}|\text{晚上/暗晡}) = \beta(W_{i-1}) \times \frac{1}{T} = 0.006744875 \times \frac{1}{2} = 0.0033725$$

$$P(\text{工作/工作}|\text{晚上/暗晡}) = \beta(W_{i-1}) \times \frac{1}{T} = 0.006744875 \times \frac{1}{2} = 0.0033725$$

補充說明：

T的個數，實際上是從辭典中統計所得到。即為統計未在 Uni-Gram 語言模型中出現過的詞目之個數，即是$C(W_i) = 0$的個數。因為每個辭典中的詞，都有可能與W_{i-1}詞連成一個 Bi-Gram 事件$Event(W_{i-1}, V_k)$。

5.8.2 實驗結果與討論

我們將$k = 4$的強化凱氏Bi-Gram模型，實際應用於客語斷詞處理。並分別做了斷詞及詞性標記的內外部測試，表五十四及表五十五分別為內部測試及外部測試的實驗結果。

表五十四：Bi-Gram 強化凱氏平滑模型內部測試結果

內部測試	Precision	Recall	F-Measure	相似度
斷詞	95.31%	94.57%	94.94%	96.63%
詞性標記	94.56%	93.82%	94.19%	-

表五十五：Bi-Gram 強化凱氏平滑模型外部測試結果

外部測試	Precision	Recall	F-Measure	相似度
斷詞	78.47%	77.25%	77.85%	82.09%
詞性標記	78.07%	76.85%	77.45%	-

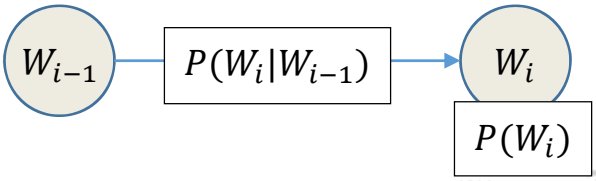
由 Uni-Gram 加成平滑法、凱氏平滑法及 Bi-Gram 的強化凱氏平滑法可觀察到，Uni-Gram 語言模型的外部測試中，加成平滑法與凱氏平滑法效果相同。而 Bi-Gram 強化凱氏平滑法在外部測試時的效果卻非常差，顯示語料有非常嚴重的稀疏問題，即便是透過 back-off 的架構，仍無法完全保持 Uni-Gram 模型本身的分佈。但使用 Bi-Gram 模型的內部測試，其 F 分數已高達 94.94%，比 Uni-Gram 模型內部測試多約 8% 的 F 分數。顯見即使 Bi-Gram 模型過於稀疏，但其所學習到的內容仍有一定的影響力。

5.9 使用 Mix-Gram 分數算法的客語斷詞

綜合以上的特性，我們試圖找出一個能保持 Uni-Gram 模型分佈，又能適時的使用到 Bi-Gram 模型中所學習到的內容，結合 Uni-Gram 及 Bi-Gram 兩個模型各自的優勢的方法。我們認為，語言模型的學習就像人類在學習語言一樣。我們有時可能會記住一些詞一起出現的用法，或是只記住一些單字。如初學某種語言一樣，即便是記住在腦海裡的詞組很少，但有學過仍能拿來用，沒有適當的詞組能用時，才用單字來敘述。因此，即便目前本論文所建置的 Bi-Gram 模型有嚴重稀疏的問題，即學習到的詞組不多，但仍有部份能在適當的時機被拿出來使用。基於這樣的概念，我們提出了 Mix-Gram 的分數算法，用混合式的候選詞序列之分數的計算方式，找出一個最佳的詞序列。

5.9.1 Mix-Gram 分數

Mix-Gram 分數算法，是混合 Uni-Gram 模型及 Bi-Gram 模型的候選詞序列的分數算法，可以組合不同的平滑法使用。我們假設一個 Bi-Gram 事件$Event(W_{i-1}, W_i)$是由一個前接詞W_{i-1}連接到後接詞W_i的連線機率$P(W_i|W_{i-1})$，而該事件的$P(W_i)$代表詞的機率，這個事件我們以圖二十一表示：

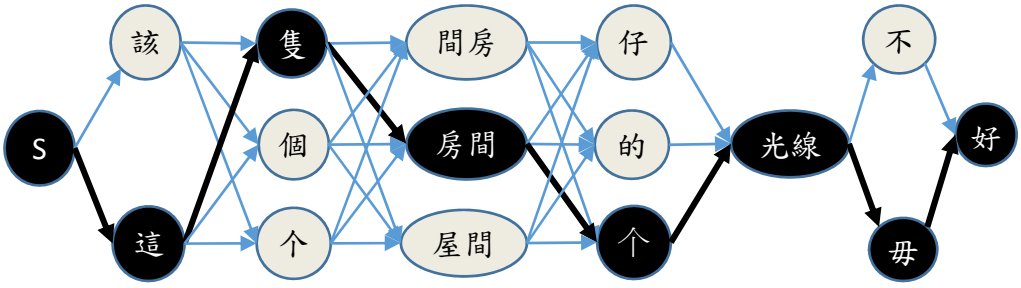


圖二十一：Mix-Gram 示意圖

其中$P(W_i|W_{i-1})$是來自 Bi-Gram 模型中所查出的機率，而$P(W_i)$是由 Uni-Gram 模型中所查出的機率。而這個事件的混合分數，計算公式為：

$$MixGram \cdot Score(S, W_1, W_2, \ldots, W_n) = \prod_{i=1}^{n} P(W_i|W_{i-1}) \times P(W_i) \quad (5\text{-}16)$$

圖二十二為一個短句的詞序列有向圖來說明 Mix-Gram 完整的計算方法：



圖二十二：Mix-Gram 候選詞序列分數計算範例

計算$MixGram \cdot Score(S, 這, 隻, 房間, 个, 光線, 毋, 好)$這個詞序列的 Mix-Gram 分數，其計算式為：

$$MixGram \cdot Score(S, 這, 隻, 房間, 个, 光線, 毋, 好) = [P(這|S) \times P(這)] \times [P(隻|這) \times P(隻)] \times [P(房間|隻) \times P(房間)] \times [P(个|房間) \times P(个)] \times [P(光線|个) \times P(光線)] \times [P(毋|光線) \times P(毋)] \times [P(好|毋) \times P(好)]$$

以上計算，當然也會遇到當$P(W_i|W_{i-1}) = 0$或是$P(W_i) = 0$的情況，也就是會出現零機率的問題。我們的處理方式如下：

$$MixGram \cdot Score(W_i|W_{i-1}) = \begin{cases} \frac{C(W_{i-1},W_i)}{\sum_{W_k} C(W_{i-1},W_k)} \times \frac{C^*(W_i)}{\sum_{W_j} C(W_j)} & if\ C(W_{i-1},W_i) > 0 \\ \gamma \times \frac{C^*(W_i)}{\sum_{W_j} C(W_j)} & if\ C(W_{i-1},W_i) = 0 \end{cases} \quad (5\text{-}17)$$

我們在$P(W_i)$上，分別測試了$\delta = 0.3$的加成平滑法及$k = 4$的凱氏平滑法，來避免零機率的問題。在$P(W_i|W_{i-1})$上則是使用一個$0 < \gamma < 1$的常數γ來避免零機率問題。為何使用常數代替？因為我們主張$P(W_i|W_{i-1})$是兩個詞連線的機率，當在 Bi-Gram 模型中未找到符合的事件，但實際上出現了，我們則給予一個γ值，來表示兩詞有可能互相連線的現象，但也許是偶然出現，因此給予很小的數值表示。但在 Bi-Gram 模型中有出現，我們採不同於 back-off 架構的平滑法，完全不分出機率，代表我們篤定兩詞之間的連線。且本身模型已經夠稀疏，應儘量保持原 Bi-Gram 所學習到的事件機率。而γ值是我們透過實驗所得到，我們發現在$\gamma = 10^{-4}$有最佳的斷詞正確率。

5.9.2 實驗結果與討論

我們實驗了「(Bi-Gram 模型)+(Uni-Gram 加成平滑模型)」及「(Bi-Gram 模型)+(Uni-Gram 凱氏平滑模型)」兩種組合的 Mix-Gram 的混合式分數計算方法，實際應用於客語斷詞處理。並分別做了斷詞及詞性標記的內外部測試，表五十六及五十七分別為「(Bi-Gram 模型)+(Uni-Gram 加成平滑模型)」的內部及外部的實驗結果。

表五十六：Mix-Gram[Bi-Gram + Uni-Gram(Additive)]內部測試結果

內部測試	Precision	Recall	F-Measure	相似度
斷詞	95.82%	95.07%	95.44%	97.17%
詞性標記	95.06%	94.33%	94.69%	-

表五十七：Mix-Gram[Bi-Gram + Uni-Gram(Additive)]外部測試結果

外部測試	Precision	Recall	F-Measure	相似度
斷詞	81.92%	80.65%	81.28%	84.84%
詞性標記	81.50%	80.22%	80.86%	-

表五十八及五十九分別為「(Bi-Gram 模型)+(Uni-Gram 凱氏平滑模型)」的內部及外部的實驗結果。

表五十八：Mix-Gram[Bi-Gram + Uni-Gram(Katz)]內部測試結果

內部測試	Precision	Recall	F-Measure	相似度
斷詞	95.76%	95.02%	95.39%	97.12%
詞性標記	95.01%	94.28%	94.64%	-

表五十九：Mix-Gram[Bi-Gram + Uni-Gram(Katz)]外部測試結果

外部測試	Precision	Recall	F-Measure	相似度
斷詞	82.05%	80.78%	81.41%	84.98%
詞性標記	81.63%	80.35%	80.99%	-

實驗結果顯示，結合 Uni-Gram 凱氏平滑模型的 Mix-Gram，在外部測試的 F 分數比結合 Uni-Gram 加成平滑模型的 Mix-Gram 高了 0.13%，因此，我們的客語斷詞模組的實作，採用此分數計算方法。

第六章 客語讀音標記及求取方法

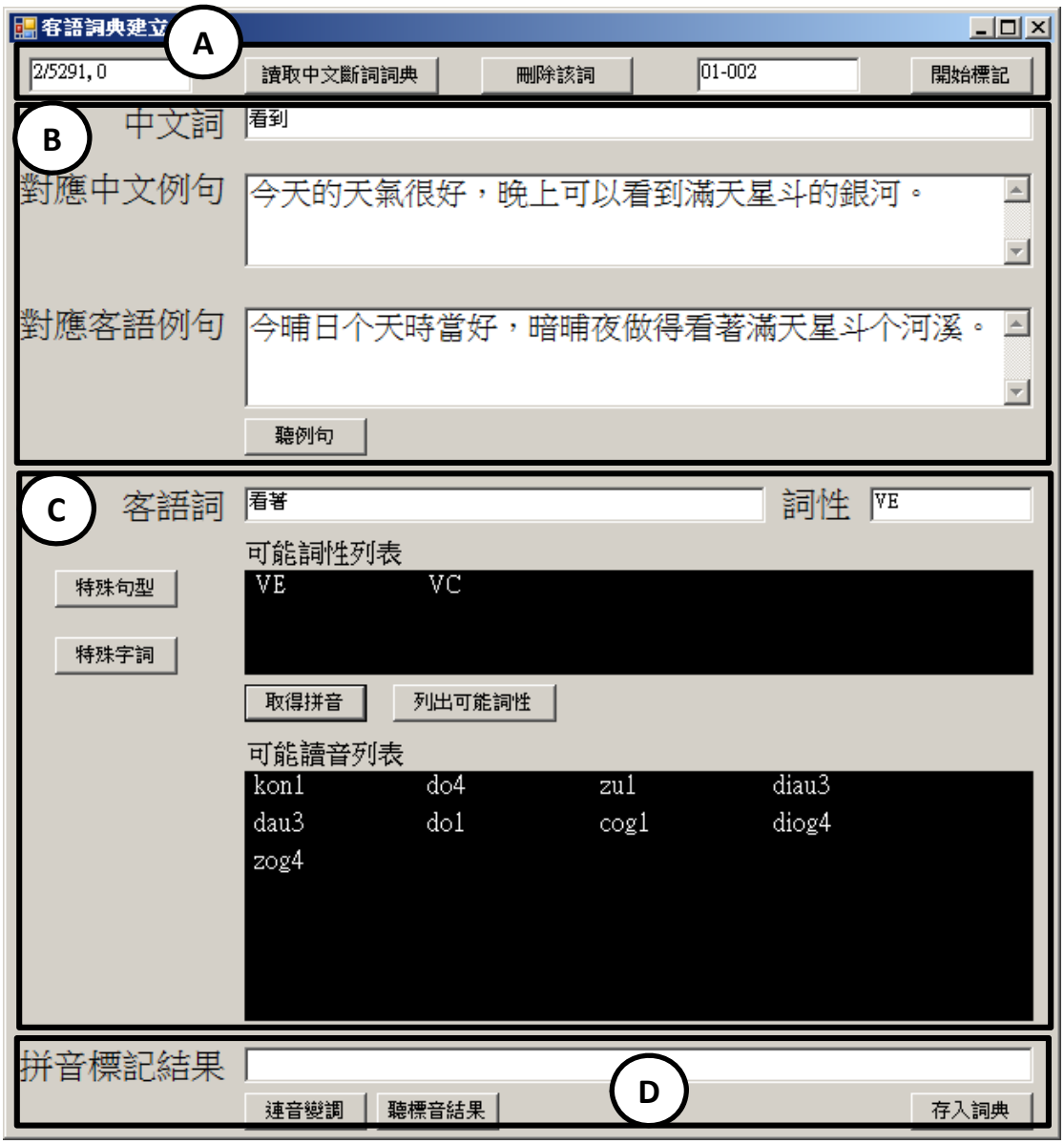
語音合成系統中，求取讀音的正確性非常重要。讀音正確與否，關係到念出來的句子是否正確。本章將介紹本論文語音合成系統中，對於讀音求取所做的探討。我們將介紹如何擴充客語讀音辭典，以及詳細說明客語讀音求取的方法及評估結果。

6.1 客語讀音的標記

我們目前僅針對國客語對照辭典中，尚未有標記上讀音的客語詞做標記。我們設計了一套標記工具，以幫助我們能快速標上正確的讀音、進行讀音辭典的擴充。本節將介紹此工具的操作流程及標記方法。

6.1.1 客語讀音標記工具介紹

針對尚未標上讀音的客語詞，因為時間及人力有限，我們並非所有的客語詞都拿來做標記。目前我們只針對「客委會初級、中級暨中高級的例句資料」中，所標記出來的國客語對應的客語詞做為標記目標。因為這份語料有句子的語音檔，可以讓我們完全正確的標記出客語讀音。圖二十三為本工具之介面，接下來我們會詳細介紹各區塊的功能以及操作方式。



圖二十三：客語讀音標記工具畫面

(A)進度框、刪除該詞、句子編號、開始標記

1. **進度框**：顯示目前進度，目前詞位置/總共詞數,剩餘未標上拼音的詞數。

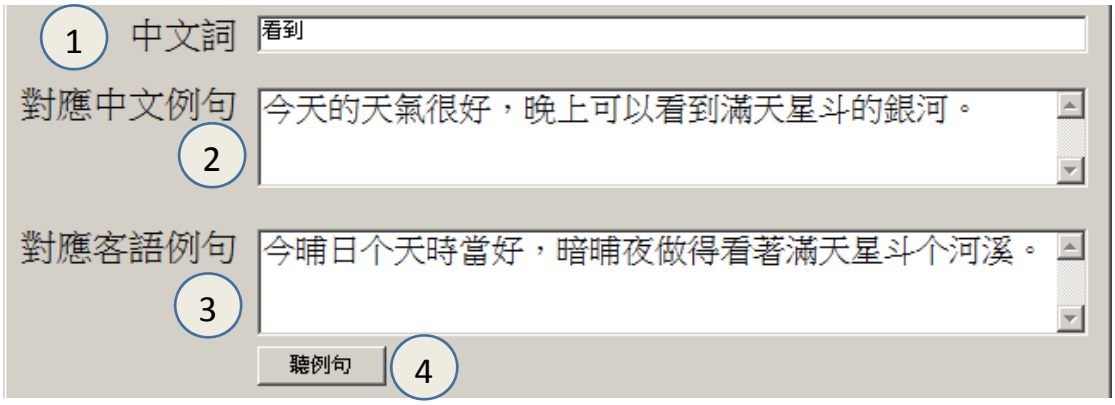
2. **刪除該詞**：不是一個正確的國客語對應詞，因為有可能標記錯了，

標記者可選擇直接刪除該詞。因此此讀音標記工具可以做雙確認的工作。

3. **句子編號**：該詞在語料中的哪句出現，依據這個編號，我們可以找出原始句子及音檔，做更嚴謹的標記。

4. **開始標記**：開始標記作業，會依據進度記錄載入要標記的詞及其例句資料。

(B) 中文詞、對應中文例句、對應客語例句、聽例句



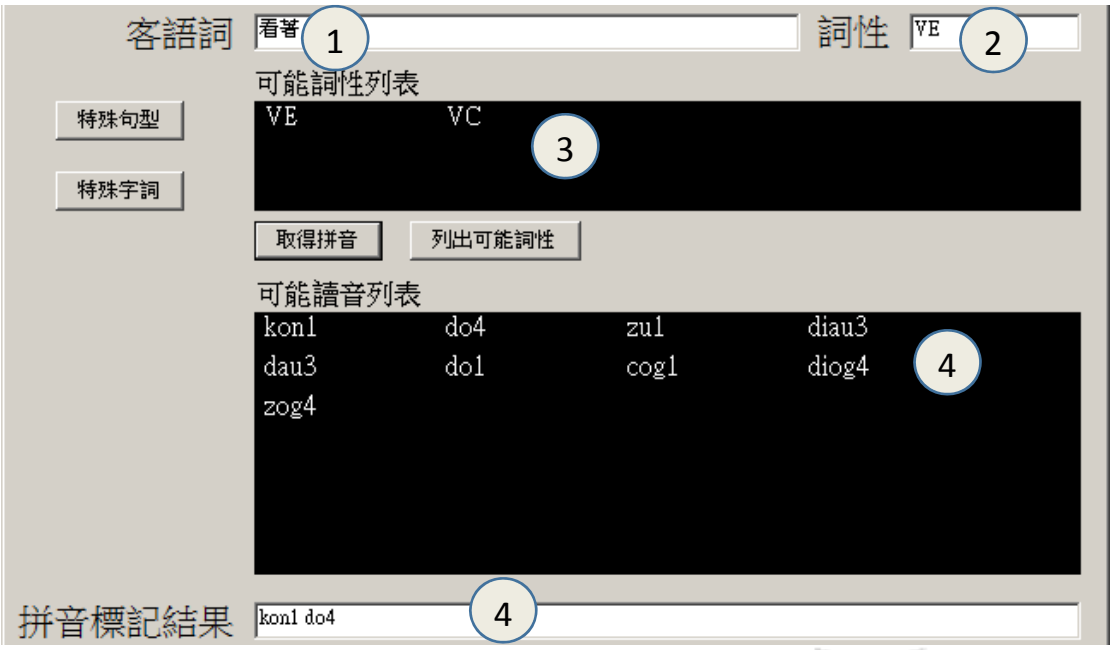
1. **中文詞**：要標記的中文詞。

2. **對應中文例句**：該中文詞所在的中文例句，讓標記者能看到其前後文及原句。

3. **對應客語例句**：該中文詞所在的客語例句，讓標記者能依據中文例句與客語例句，來判斷其客語詞及讀音。

4. **聽例句**：可聽取該句客語例句的原始語音檔，以判斷讀音該如何標記。

(C) 客語詞、詞性、可能詞性列表、可能讀音列表



1. **客語詞**：中文詞所對應的客語詞。

2. **詞性**：該客語詞，所對應的中文詞之詞性。

3. **可能詞性列表**：

   a. 「列出可能詞性」按鈕，可顯示該中文詞所有可能的詞性。資料來源是中文斷詞辭典。

   b. 以***滑鼠右鍵***點選列表上的詞性，「2.詞性」的欄位改為該詞性。

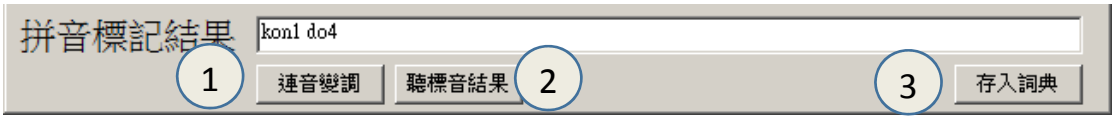
4. **可能讀音列表**：

   a. 按「取得拼音」按鈕，可找出客語詞每個音節，其所有可能的拼音。

   b. 以***滑鼠右鍵***點選列表上的讀音，「4.拼音標記結果」的欄位，即

會列出點選的拼音結果，即為標記的結果。

(D)連音變調、聽標音結果、存入辭典

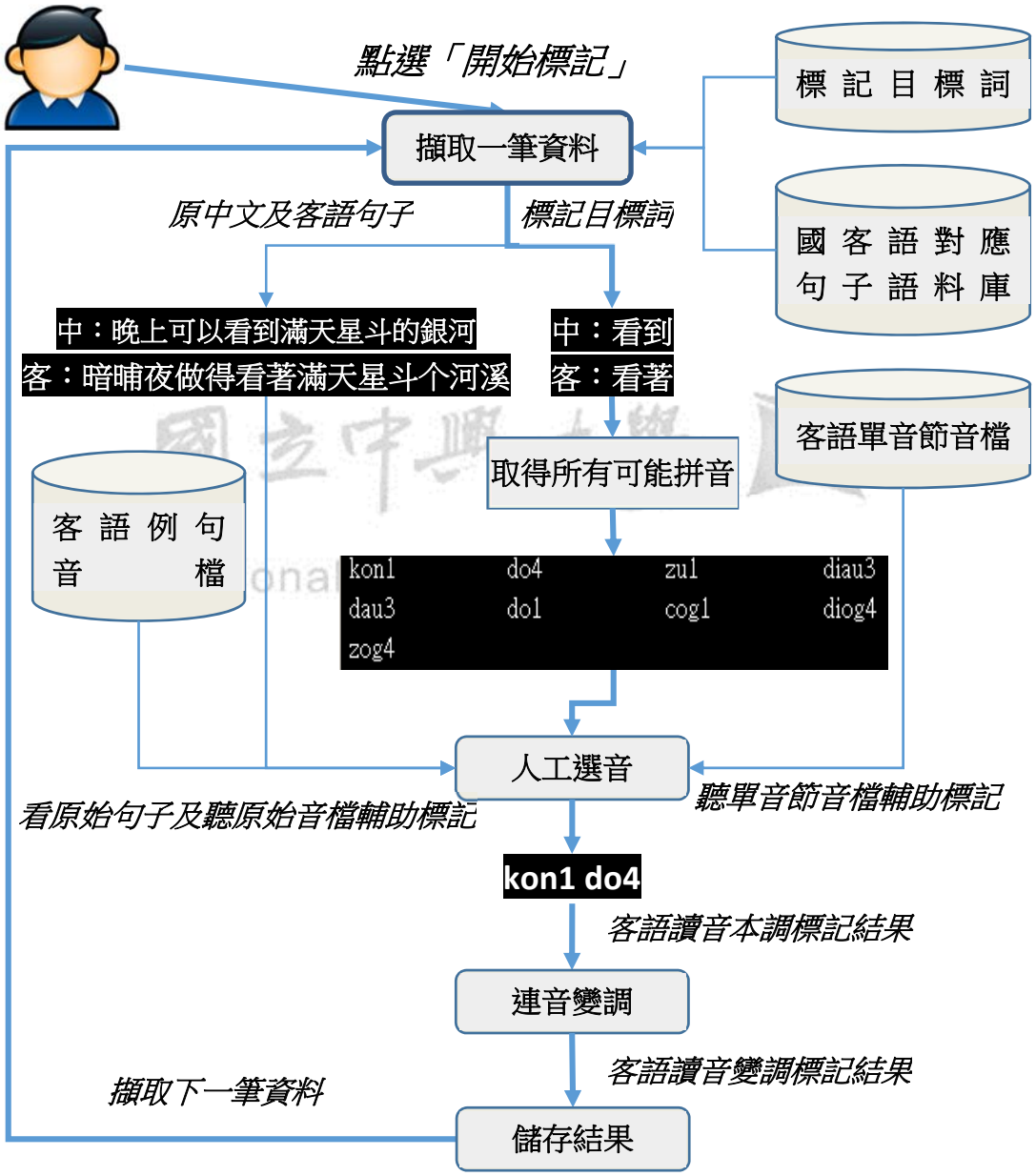


1. **連音變調**：按「連音變調」，會將「拼音標記結果」上的拼音，做四縣腔的連音變調。

2. **聽標音結果**：可聽取讀音標記後的結果。

3. **存入詞典**：將讀音標記結果，儲存到辭典中。

我們舉一個例子，來說明整個標記的流程。圖二十四為標記「看到/看著」讀音的示意圖。這組國客語詞的對應，是來自於句子「晚上可以看到滿天星斗的銀河」，因此工具會將該句的國、客語句子也呈現在畫面上，並且提供語音檔讓標記者聽取整句的讀音，輔助其標記作業。



圖二十四：客語讀音標記工具程式運作程圖

6.1.2 客語讀音標結果

我們從辭典中，挑出來自「客委會初級[14]、中級暨中高級[12][13]」語料所標記出來且未標上讀音的客語詞，共 5291 個做為標記讀音的目標。標記完成後，將這些資料與原有的客語發音辭典合併，並做全面性的人工校正、過濾掉有問題的資料(亂碼、缺音、欄位資料錯置)。最後我們得到一個經人工校正、有 32453 筆讀音資料的客語詞彙發音辭典(表六十)。其資料樣貌及分佈如下，圖二十五中，分別是：客語詞、詞的拼音、詞性：

```
32447 私家    sii3 kia2   Na
32448 捉包    zog4 bau2   Na
32449 煠  sab1    VC
32450 个  ge1 T
32451 吾  nga2    DE
32452 捉着    zug1 do4    VC
32453 朝晨    zeu2 siin3  Nd
```

圖二十五：客語詞彙發音辭典資料樣貌

表六十：客語詞彙發音辭典分佈統計表

字詞	總數
一字詞	2175
二字詞	19190
三字詞	6685
四字詞	3947
五字詞	301
六字詞	81
七字詞	65
八字詞	9
總計	32453

6.2 實驗資源及評估方法

本節將介紹客語讀音求取所使用到的訓練及測試資料，以及詳細說明我們讀音求取的評估方法。

6.2.1 實驗資源

我們客語發音辭典的主要來源有：(一)客委會初級[14]、中級暨中高級認證語料[12][13]、(二)台北市客委會-現代客語詞彙彙編，這兩大來源有標記上客語讀音的資料。我們將原有的客語詞彙發音辭典與新標記的客語讀音合併、人工校正後，得到一個有 32453 個詞的客語詞彙發音辭典。並將辭典分割出七成作為訓練集，三成作為測試集。使用分佈如下表六十一所示：

表六十一：客語讀音語料的使用分佈

	訓練資料集	測試資料集
詞數	22718	9735
字數	55374	23542
讀音數	55382	23548

6.2.2 評估方法

我們採用整個詞的讀音完全正確才給分的方式，正確率的計算如下：

$$正確率 = \frac{系統正確求出讀音的詞數}{參考答案的總詞數} \quad (6\text{-}1)$$

我們將測試語料的客語詞，輸入到讀音求取的演算法當中，得到讀音後再與參考答案做比對。最後統計總共對了幾個詞，並計算其正確率。表六十二及六十三是判斷正確與否的例子：

表六十二：客語讀音求取評估範例一

輸入詞	討夫娘
標準答案	to4 bu2 ngiong3
系統輸出	to4 fu2 ngiong3
正確與否	Wrong

表六十三：客語讀音求取評估範例二

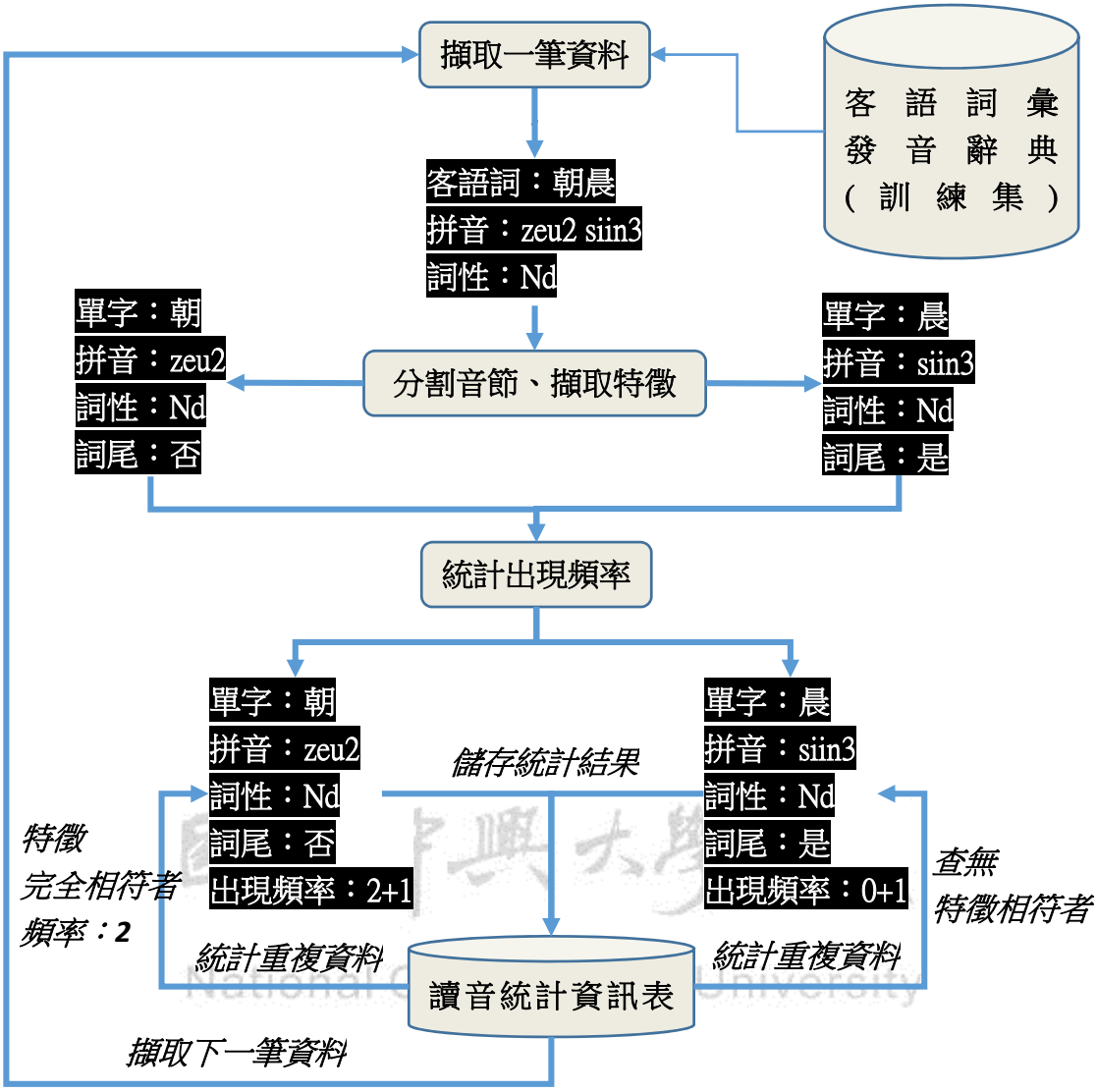
輸入詞	寒流
標準答案	hon3 liu3
系統輸出	hon3 liu3
正確與否	Correct

6.3 讀音求取方法

我們讀音求取的方法，是採用以詞或字(Word-Based)找出對應的讀音。我們建立了一個讀音資訊統計表，來輔助讀音的選取。求取方法是以發音辭典為主，讀音資訊統計表為輔。本節將介紹我們如何建置這個讀音統計資訊表，以及讀音求取的演算法。

6.3.1 建立讀音統計資訊表

我們從客語詞彙發音辭典中所切出的訓練集，將所有的客語詞分割成單字，並擷取每個字的「讀音、詞性、是否為詞尾」等特徵做統計，所有的特徵要相符才視為同一筆，即便是讀音、詞性一樣，是否為詞尾的特徵不同，也算不同筆。因此只要任何一個特徵不同，我們則記為不同筆資料。圖二十六為訓練流程的示意圖。



圖二十六：讀音統計資訊表訓練流程

訓練完的讀音資訊統計表如表六十四所示,我們記錄了五個欄位，分別是「客語單字」、「讀音」、「詞性」、「是否詞尾」、「頻率」。

表六十四：單字「院」的讀音統計資訊表範例

客語單字	讀音	詞性	詞尾	頻率
啾	jio4	Na	是	**1**
啾	jiu2	VH	否	**2**
啾	jiu3	Na	否	**3**
啾	lio2	Na	否	**1**

我們除了針對全部的訓練資料集，做讀音資訊統計表的訓練外。也以相同的訓練方式，額外的建立出破音字的讀音資訊統計表。這張表針對了表六十五所列舉得破音字，做統計、訓練：

表六十五：客語多音字與其可能讀音列表

多音字	可能讀音					
行	hang3	hong3	hen1	hen3		
調	tiau3	tiau1	diau1			
重	cung1	ciung3	cung2	qiung3	dong1	
差	ca2	cai2				
易	i1	id1				
口	kieu4	heu4				
著	cog1	dau3	diau3	do4	zog4	zu1
合	kab4	hab1	gag4	gab1	gab4	
落	lag1	lau1	log1	lab4		
背	boi1	ba3	bi1	poi1		
中	dung2	zung2	zung1			
正	zang2	zang1	ziin1			

6.3.2 讀音求取演算法

我們的客語讀音求取方法，輸入是以一個詞為單位。其求取流程以客語詞彙發音辭典的讀音為優先，先查詢發音辭典，若有找到則直接使用詞的讀音。若查不到詞的讀音，才將客語詞分割為單字，先查詢破音字讀音資訊統計表，若查不到，則改查詢讀音資訊統計表。若上述的詞典、統計表都查不到讀音時，則查詢客語單音節發音辭典。以下是客語讀音求取完整的演算法流程：

Step1. 搜尋「客語詞彙發音辭典」

1. 使用「詞 + 詞性」查詢

a. 若有符合特徵的客語詞，則使用其讀音並跳到 ***Step5*** 結束流程。

b. 若沒有，則進行第 ***2*** 程序。

2. 使用「詞」查詢

a. 若有符合特徵的客語詞，則使用其讀音並跳到 ***Step5*** 並結束流程。

b. 若沒有，則進行 ***Step2*** 的程序。

Step2. 將客語詞拆成單字，並將每個單字搜尋「破音字讀音統計資訊表」

1. 使用「詞性 + 詞尾」特徵查詢，並選擇頻率最高者。

a. 若有符合特徵的客語字，則使用其讀音並繼續處理下一個單字，直到全部處理完畢，跳到 ***Step5*** 並結束流程。

b. 若沒有，則進行第 ***2*** 程序。

2. 使用「詞性」特徵查詢，並選擇頻率最高者。

a. 若有符合特徵的客語字，則使用其讀音並繼續處理下一個單字，直到全部處理完畢，跳到 ***Step5*** 並結束流程。

b. 若沒有，則進行第 ***3*** 程序。

3. 使用「詞尾」特徵查詢，並選擇頻率最高者。

a. 若有符合特徵的客語字，則使用其讀音並繼續處理下一個單字，直到全部處理完畢，跳到 ***Step5*** 並結束流程。

b. 若沒有，則進行 ***Step3*** 程序。

Step3. 將客語詞拆成單字，並將每個單字搜尋「讀音統計資訊表」

1. 使用「詞性 + 詞尾」特徵查詢，並選擇頻率最高者。

a. 若有符合特徵的客語字，則使用其讀音並繼續處理下一個單字，直到全部處理完畢，跳到 ***Step5*** 並結束流程。

b. 若沒有，則進行第 ***2*** 程序。

2. 使用「詞性」特徵查詢，並選擇頻率最高者。

a. 若有符合特徵的客語字，則使用其讀音並繼續處理下一個單字，

直到全部處理完畢，跳到 ***Step5*** 並結束流程。

b. 若沒有，則進行第 ***3*** 程序。

3. 使用「詞尾」特徵查詢，並選擇頻率最高者。

a. 若有符合特徵的客語字，則使用其讀音並繼續處理下一個單字，直到全部處理完畢，跳到 ***Step5*** 並結束流程。

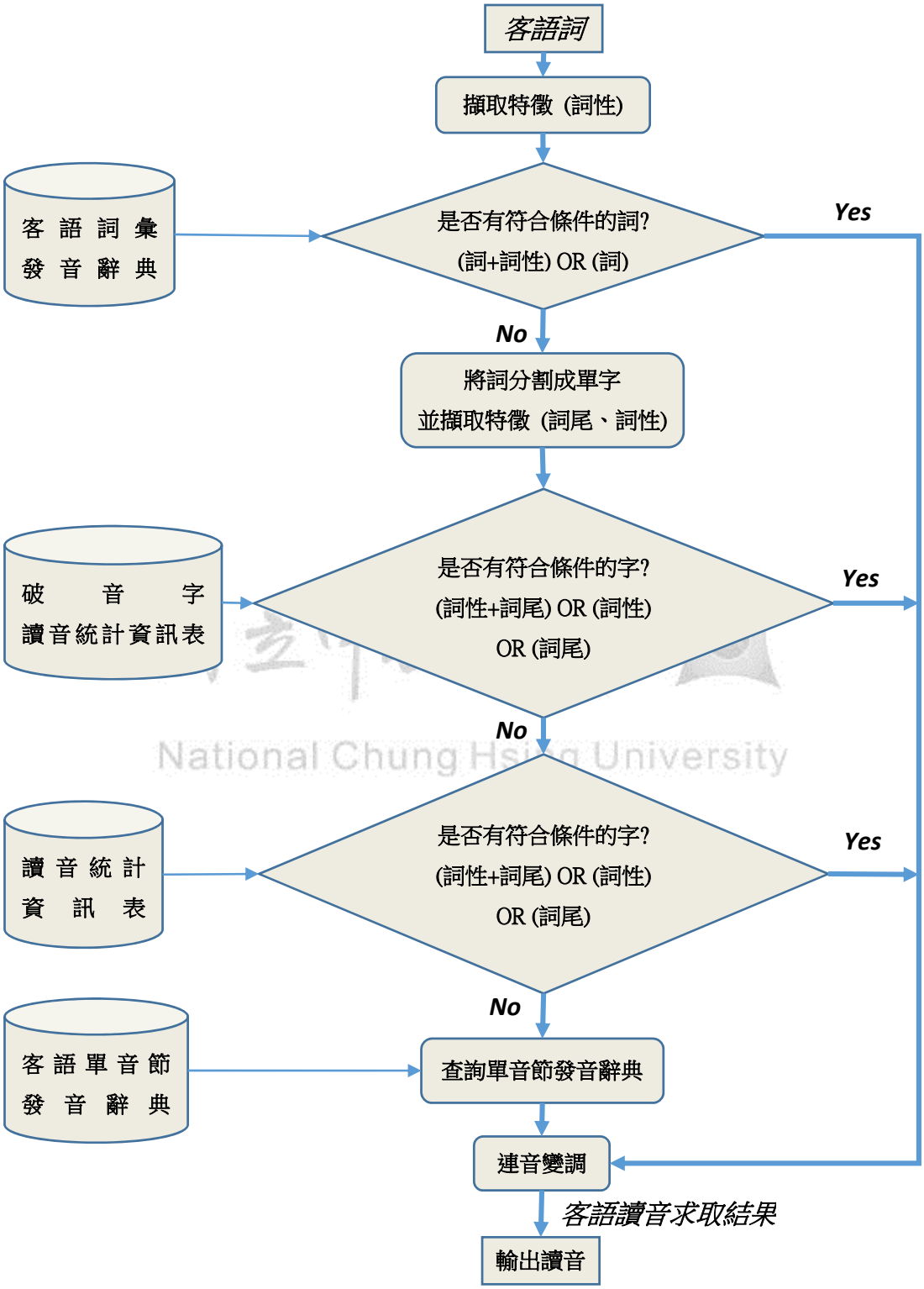
b. 若沒有，則進行 ***Step3*** 程序。

Step4. 將客語詞拆成單字，並將每個單字搜尋「客語單音節發音辭典」

上列步驟都找不到客語讀音時，直接查詢單音節發音辭典。且繼續處理下一個單字，直到全部處理完畢，跳到 ***Step5*** 並結束流程。

Step5. 連音變調，並輸出讀音求取結果。

圖二十七為整個讀音求取演算法的執行流程：



圖二十七：客語讀音求取演算法流程圖

6.4 實驗結果

表六十六為上述讀音求取流程的內外部測試結果，由實驗結果顯示，使用讀音統計資訊表於基本的客語讀音求取，已有不錯的結果。

表六十六：客語讀音求取實驗結果

內部測試	89.43%
外部測試	82.81%

未來我們可以增加客語詞彙發音辭典的量，並且可以針對破音字的處理做更進一步的改善,結合本實驗室客語讀音各種方法的分類器,找出關鍵的特徵來訓練破音字模型、提升破音字的處理效能。

第七章 中文轉客文語音合成系統實作

本論文的語音合成系統，是以模組化的方式開發。我們將不同功能的模組分別開發、包裝成動態連結函式庫(Dynamic-link Library)，再將這些模組組合成系統。系統的開發環境及使用工具如下表所示：

表六十七：系統開發環境與工具

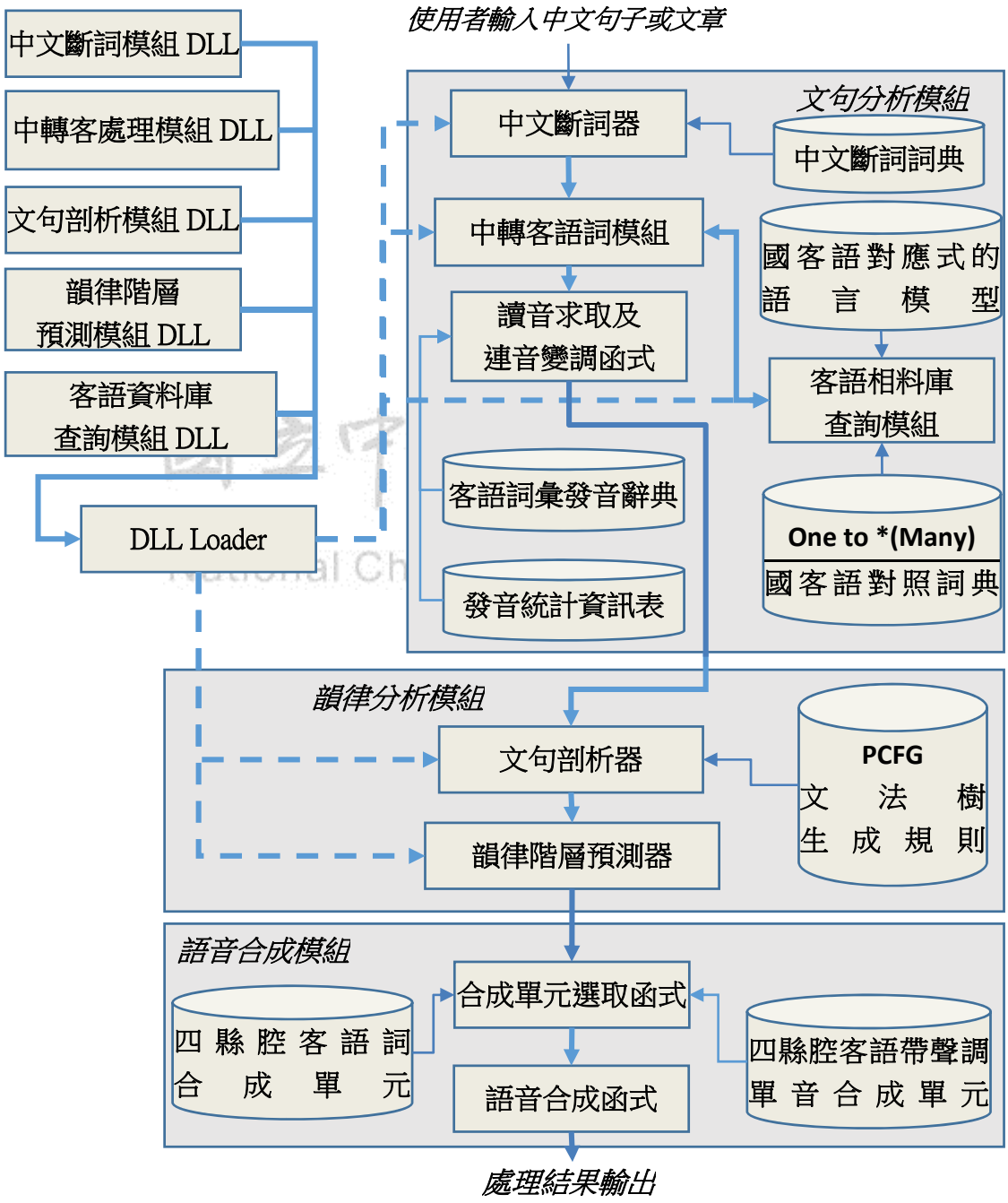
Software & Toolkit	Version
Operation System	Windows 7
Database	MySQL 5.0.51a
Programming	Visual C++ 2005 Visual C++ 2012 Visual C# 2012 Eclipse Java
DBMS	phpMyAdmin - 2.10.3
Boost	1.41
IKVM	7.2.4630

1. **中文斷詞模組**：使用 Visual C++ 2005 及 Boost 開發。
2. **中文詞轉客語詞模組(客語斷詞處理)**：使用 Visual C# 2012 開發。
3. **文句剖析模組**：使用 Eclipse JAVA 開發。
4. **韻律階層預測模組**：使用 Visual C++ 2012 開發。
5. **客語資料庫查詢模組**：使用 Visual C++ 2005 及 Boost 開發。
6. **讀音求取模組、單元選取模組、語音合成模組**：使用 Visual C# 2012 開發。
7. **語音合成系統整合介面**：使用 Visual C# 2012 及 IKVM 開發。

7.1 系統架構與運作流程

7.1.1 系統架構

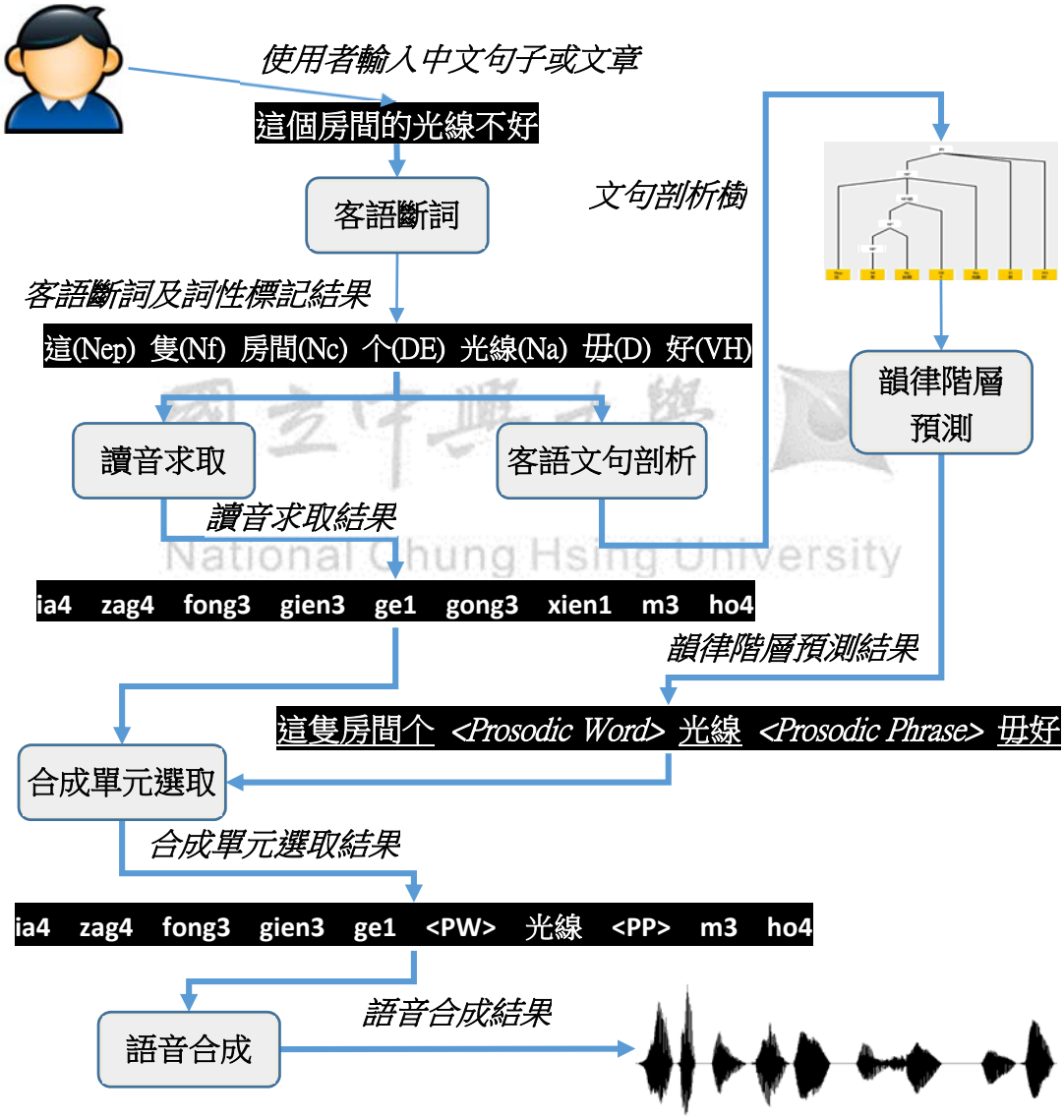
系統模組間的關係及系統架構圖，如圖二十八所示：



圖二十八：2014 興大客語語音合成系統模組關係及系統架構圖

7.1.2 運作流程

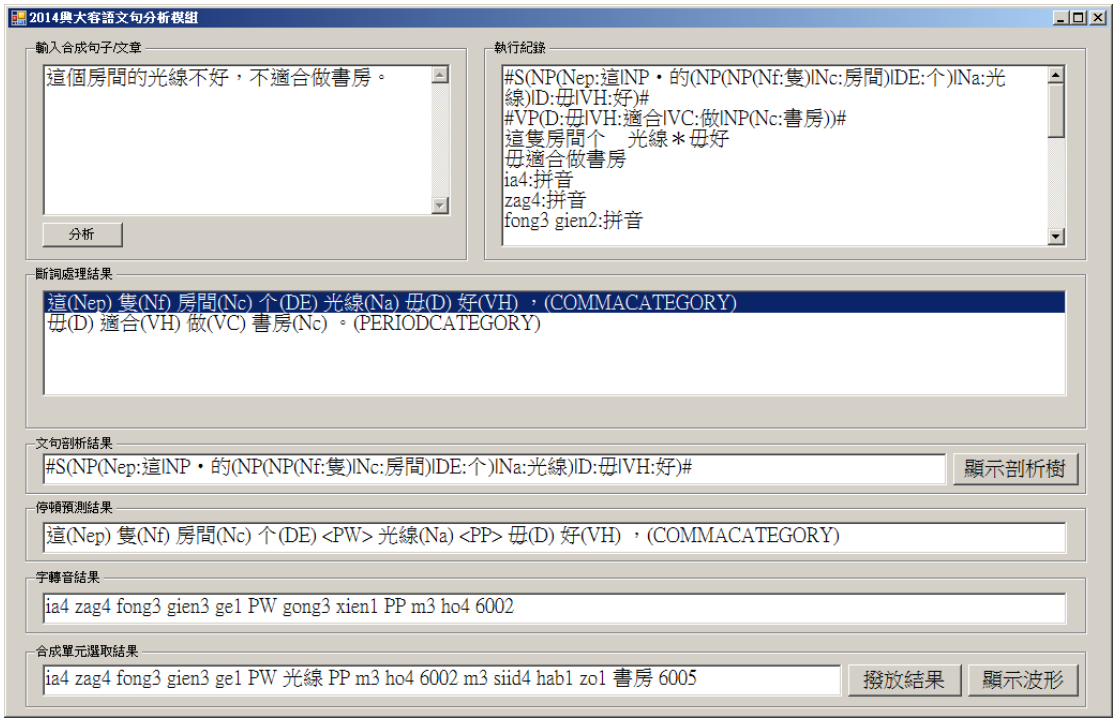
圖二十九是本論文的中文轉客文語音合成系統之系統運作流程，我們以中文短句「這個房間的光線不好」為例，輸入至系統並呈現每個階段完整的輸入及輸出結果。



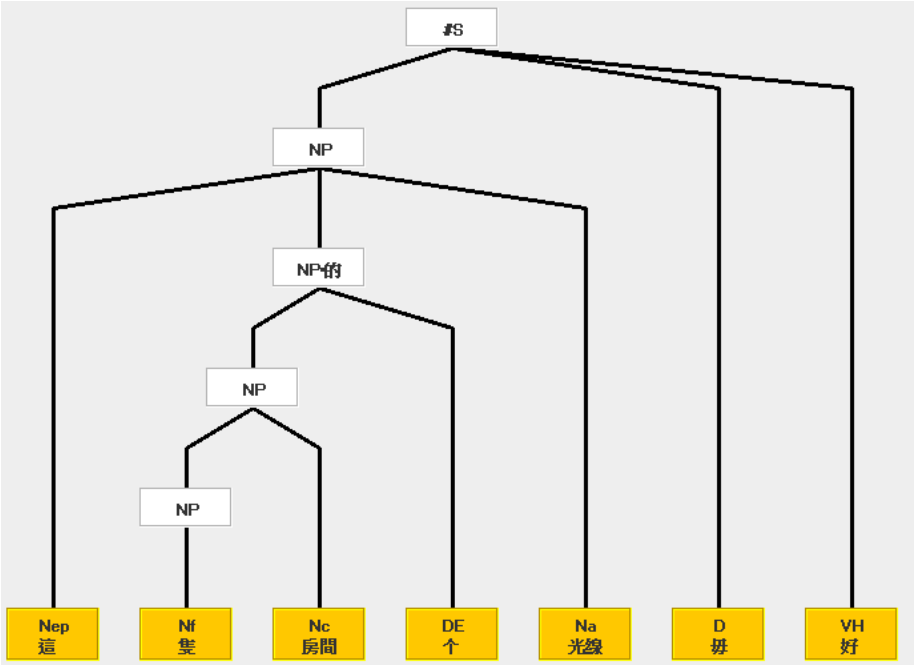
圖二十九：2014 興大客語語音合成系統系統運作流程圖

使用者輸入中文句子或文章後，系統會先將輸入資料做客語斷詞

的處理，得到的客語斷詞及詞性標記結果，再分別進行客語讀音的求取及客語文句的剖析。經讀音求取後的客語句子會輸出整句的讀音；經剖析後的客語句子，會輸出句子的剖析樹結果。剖析樹結果會送入韻律階層預測模組中，自動標記出韻律片語(Prosodic Phrase)或韻律詞(Prosodic Word)的邊界。最後將讀音求取結果及韻律標記結果送入合成單元選取模組，並選出最適合的合成單元，送入語音合成器中將這些單元合成出來，最後輸出客語語音。圖三十及圖三十一，分別為本系統的使用者介面及客語文句剖析結果：



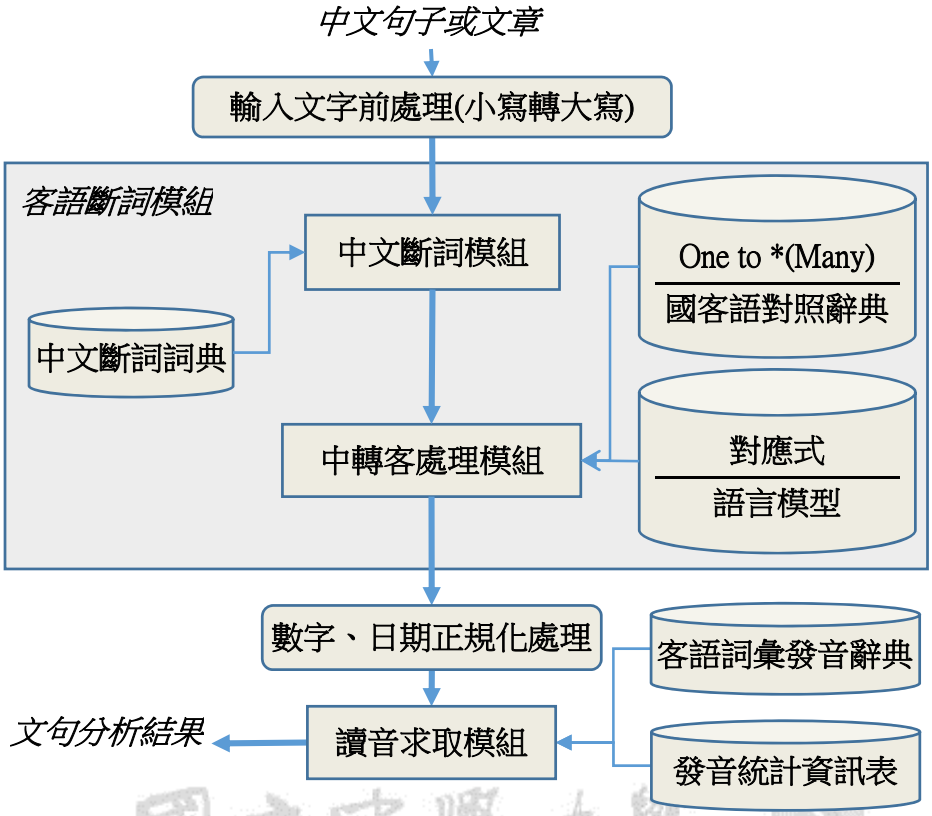
圖三十：2014 興大客語語音合成系統操作介面



圖三十一：2014 興大客語語音合成系-客語文句剖析結果畫面

7.2 文句分析模組

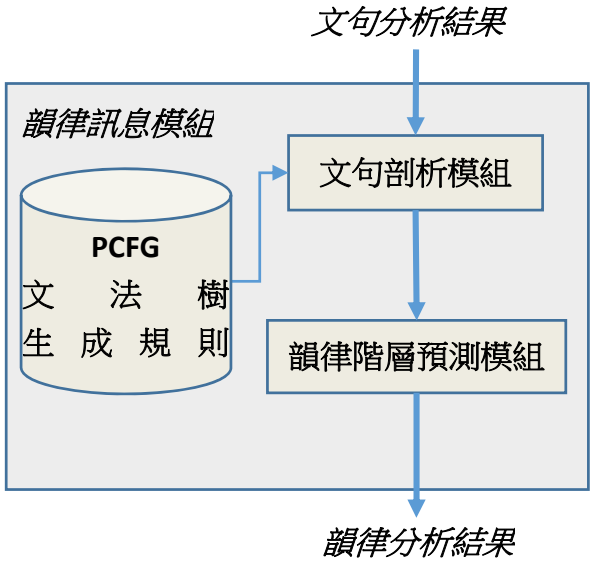
本系統的文句分析模組，是由三個不同功能的子模組所組成。分別是 1.中文斷詞模組、2.中轉客處理模組、3.讀音求取及連音變調模組。其中 1、2 項我們開發成一個客語斷詞模組。文句分析模組是語音合成系統當中最前端且重要的模組，它負責分析輸入到系統中的文句所包含的 1.詞及詞性、2.文句文法及 3.每個字的讀音甚至是 4.文意，等非常重要的文脈訊息。本系統的文句分析模組除了文意訊息外，包辦了前 1 到 3 項的功能，使用者輸入「中文句子或文章」到系統中，我們能分析出內含的各種訊息。圖三十二文句分析模組的架構圖：



圖三十二：文句分析模組架構圖

7.3 韻律訊息模組

本系統的韻律分析模組，是由兩個不同功能的子模組所組成。分別是 1.文句剖析器、2.韻律階層預測模組。文句剖析器負責將斷詞及標記詞性後的客語句子做文法的剖析，並輸出文法剖析樹。韻律階層預測模組再根據文法剖析樹結果，來找出句子中的韻律階層，其中包括 1.韻律片語(Prosodic Phrase)、2.韻律詞(Prosodic Word)。我們再根據這些韻律標記，給予適當的停頓訊息。圖三十三為韻律訊息模組的架構圖：



圖三十三：韻律訊息模組架構圖

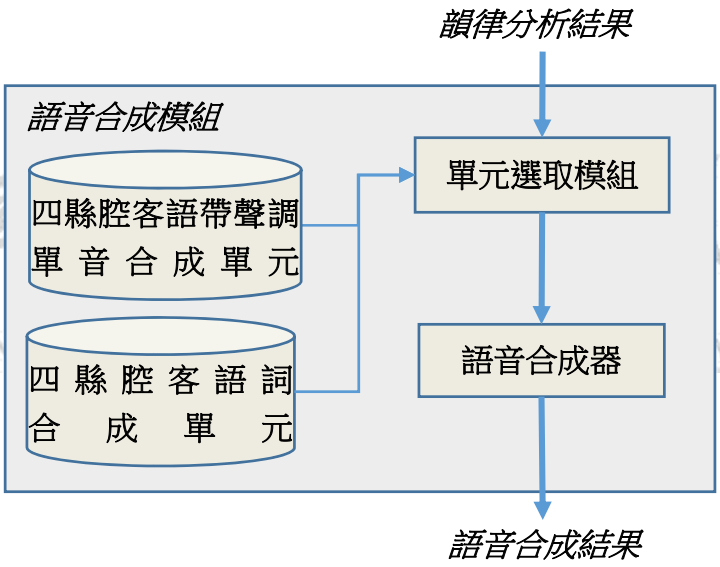
表六十八是我們針對標點符號及韻律階層所制訂的停頓時長：

表六十八：各種停頓類型的時長

停頓類型	停頓時長(ms)
小停頓(Minor break，PW)	200
中停頓(Major break，PP)	350
逗號(，)	550
句號(。)	650
問號(？)	650
驚嘆號(！)	650
分號(；)	600
頓號(、)	400
冒號(：)	450

7.4 語音合成模組

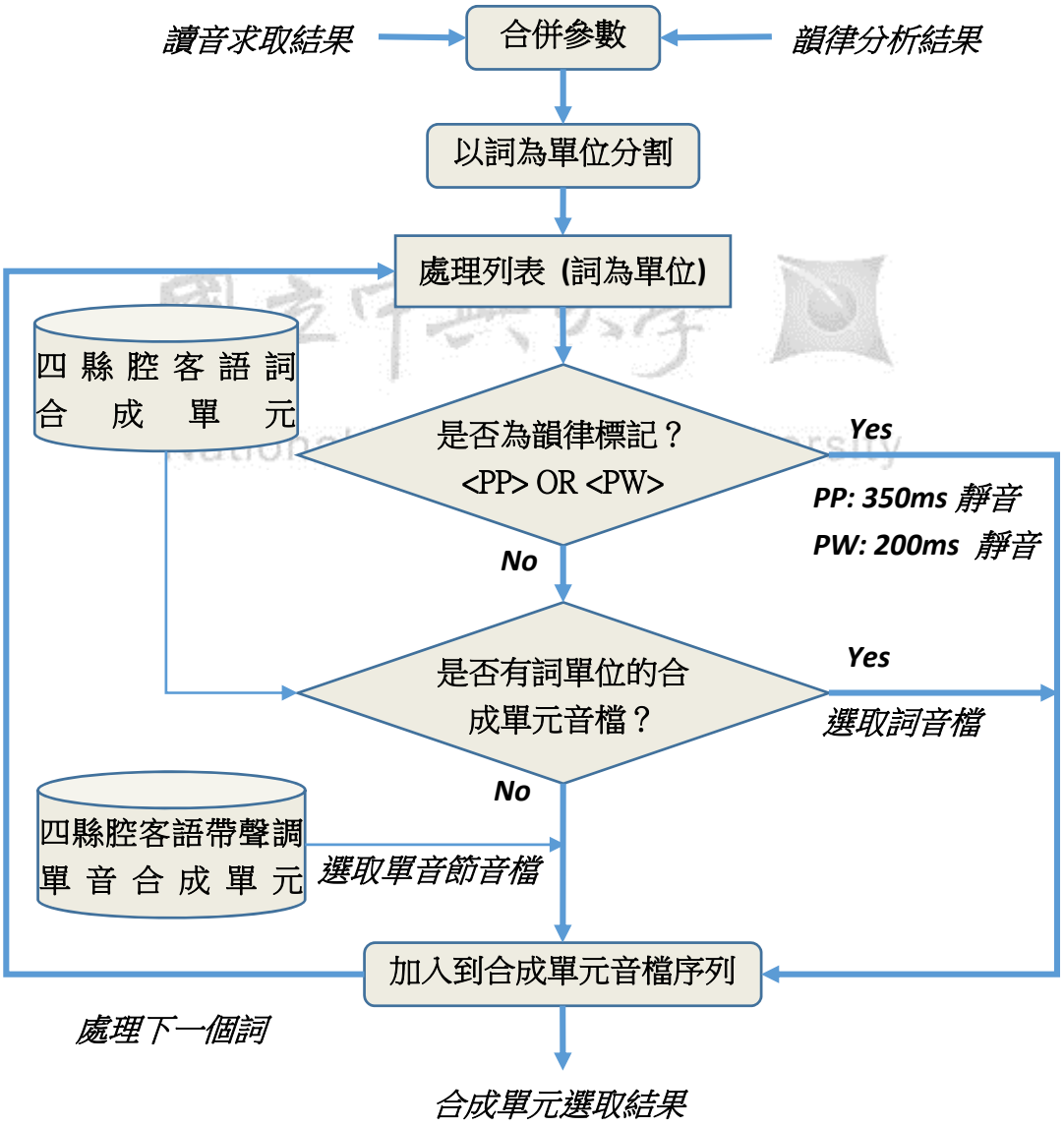
本系統的語音合成模組，是由兩個不同功能的子模組所組成。分別為 1.單元選取模組、2.語音合成器。單元選取模組負責接收韻律分析及讀音求取的結果，從詞合成單元及單音節合成單元語音資料庫中，找出適合的合成單元。而語音合成器則負責將這些選取出來的語音檔序列做串接合成。圖三十四為語音合成模組的架構圖：



圖三十四：語音合成模組架構圖

7.4.1 單元選取模組

因為我們的合成單元並非以句子語音語料所切出，而是採用以詞為單位錄製的。因此我們並未訓練出合成單元選取模型來做單元選取，而是採用 Word-Based 的方法，以詞或音來找出合成單元。合成單元選取的流程如下圖：

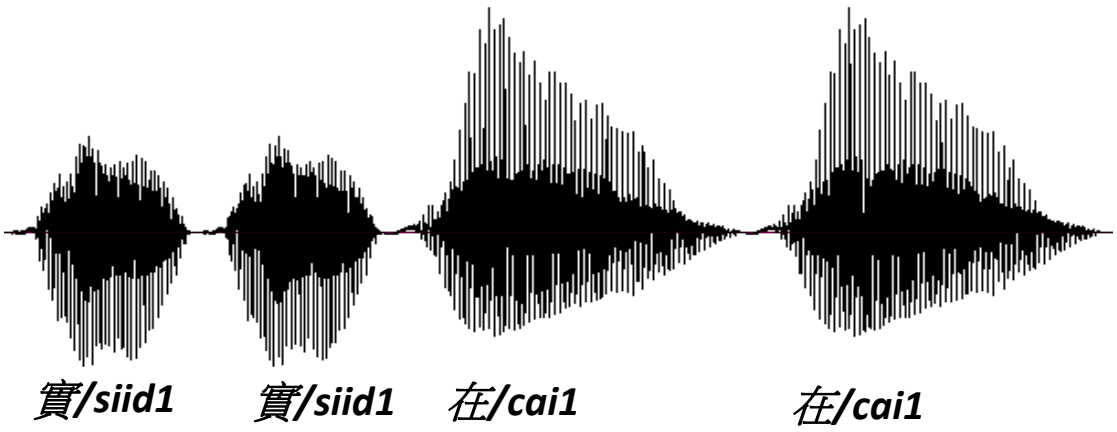


圖三十五：單元選取模組-合成單元選取流程圖

7.4.2 語音合成器

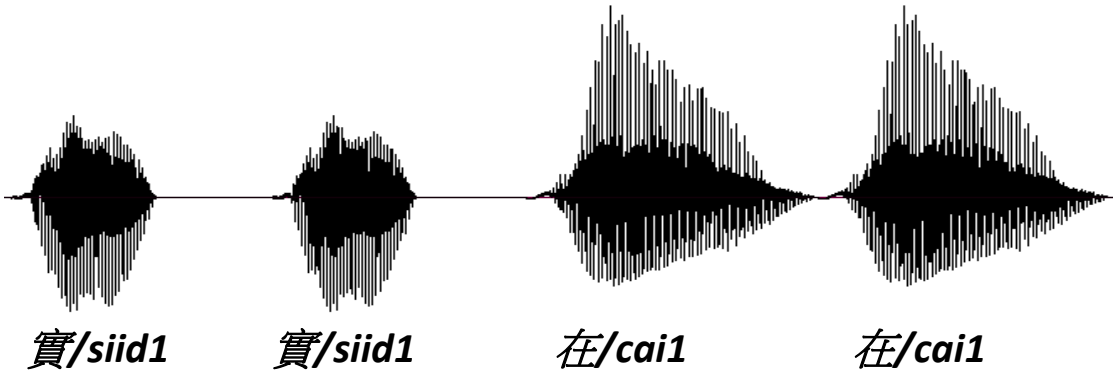
本論文採用串接合成法來做語音合成，而我們使用的單音節合成單元皆是帶聲調的合成單元，因此我們不對合成單元做音長(Duration)及音高(Pitch)的調整。我們針對客語 670 種單音節，錄製了四種聲調的合成單元。而客語入聲字部分，我們也錄製了兩種聲調。除了單音節合成單元外，我們也錄製了詞為單位的合成單元，錄製範本包括客委會初級[14]、中級暨中高級的詞彙[12][13]，以及國客語對照辭典中常用的客語詞，近期也新增了中國大陸部分地名及台灣各縣市地名…等。

對於合成結果的觀察,我們發現入聲字的音檔有音長過短的問題，容易造成合成出來的語音太短促、無法聽懂的情況。因此我們針對入聲字有加入 100ms 的靜音。圖三十六是未加入靜音的例子：



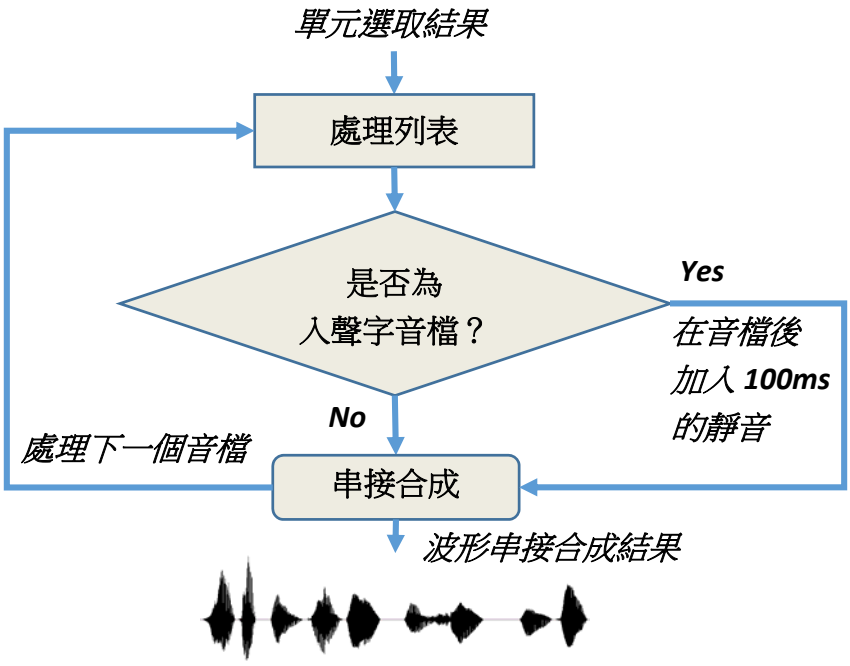
圖三十六：客語「實實在在」未加入靜音的合成波形

其中「實」為客語入聲字，其音節時長為 190ms，若沒有加入靜音，實際聽起來有短而急促的感覺，會有聽不太懂的情況。圖三十七是我們在入聲字「實」音節的後面，均加入 100ms 靜音後的波形：



圖三十七：客語「實實在在」加入靜音的合成波形

實際聽測觀察，原本聽不太懂的現象已有改善。圖三十八是我們語音合成器的流程圖：



圖三十八：語音合成器之串接合成流程圖

7.5 聽測實驗

本論文採用平均主觀分數(Mean Opinion Score, MOS)方法，來測驗合成出的語音自然度及文意的正確性。合成方法，分別由舊系統[34]及本論文系統合成出句子，並在本系統合成出的句子中，在詞內分別加入了 0、0.05、0.10、0.15 秒的靜音。

7.5.1 實驗語料

本論文所使用的合成單元語料，是委由陳婷芳老師所錄製。該語料是以一個音節或一個詞為單位錄製,我們共錄製了 2427 個單音節合成單元，以及 3392 個詞合成單元。這些合成單元音檔錄製格式為：44.1kHz、16 bits，儲存成 Windows PCM 格式(wav 檔)。

7.5.2 實驗環境

我們將實驗分為兩組，主要是測試不熟悉客語者與熟悉客語者，對於本系統所輸出的語音之評分結果。因為在評分時，會受到念對與否和客語聽力能力所影響。

第一組實驗的受測者，不是以客語作為主要母語的八位實驗室同學，在實驗室以頭戴式耳機作測驗。測試時環境安靜無聲，因此不會受到背景噪音影響。

第二組實驗，受測者本身是以客語為主要母語的四位同學及兩位客家人士，也是以頭戴式耳機作測驗。

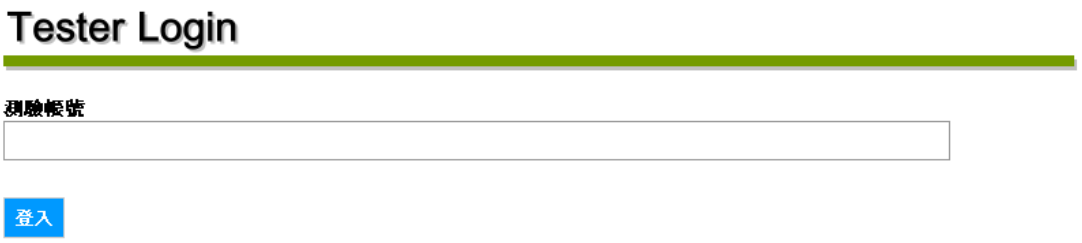
7.5.3 線上聽測

本實驗室開發了一個可進行線上聽測實驗的系統，該系統提供了更便利的測驗環境及操作介面，能有效率的作大量聽測資料的收集。研究者只需將音檔壓縮包裝成一個 Zip 格式的檔案，上傳到該系統中，並設定測驗主題、合成方法名稱以及受測人數，即可建立出受測者聽測之虛擬帳號，如圖三十九、四十所示。

圖三十九：研究者建立聽測實驗表介面 1

圖四十：研究者建立聽測實驗表介面 2

將產生的虛擬帳號分配給受測者後，每位受測者即可登入受測(圖四十一)。受測者亦可線上暫存聽測結果，也可重複聽取每句音檔，若全部語音都評分完成後，按下儲存結果即可送出聽測分數(圖四十二)。



圖四十一：線上聽測登入畫面

Listening Test

聽測主題：2014興大客語語音合成系統-聽測實驗

受測者1 (0303171) 登出

序號	句子	評分
1	企業界能突破「同行是冤家」的觀念	1 2 3 4 5
2	民進黨要求重組中央選委會	1 2 3 4 5
3	隨著鼓聲、口哨聲和森巴等音樂聲	1 2 3 4 5
4	海洋生態環境的污染越來越嚴重	1 2 3 4 5
5	二人差點發生肢體衝突	1 2 3 4 5
6	開闢山區公路是輕而易舉的事	1 2 3 4 5
7	隨著鼓聲、口哨聲和森巴等音樂聲	1 2 3 4 5
8	營建業工人的薪資因地域也有差異	1 2 3 4 5
9	倉庫原設計爲放置非危險非易燃物品	1 2 3 4 5
10	倉庫原設計爲放置非危險非易燃物品	1 2 3 4 5
11	公路兩旁的建築式樣單調	1 2 3 4 5
12	民眾平日必須步行一公里以上才能取得用水	1 2 3 4 5
13	公路兩旁的建築式樣單調	1 2 3 4 5

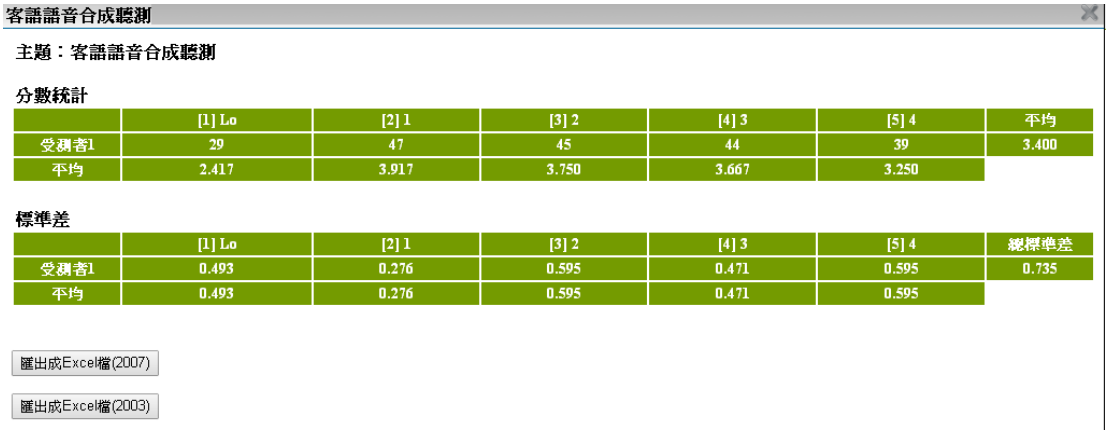
圖四十二：線上聽測評分畫面

最後，研究者只需登入後端系統介面，即可看到每句的評分結果(圖四十三)，以及整體的平均分數(圖四十四)。

受測者1 (1924121) 完成聽測

序號	ID	句子	檔案	方法	分數
1	119	隨著鼓聲、口哨聲和森巴等音樂聲	11.wav	4 (方法 5)	3
2	117	開闢山區公路是輕而易舉的事	09.wav	4 (方法 5)	3
3	111	民眾不要冒然嘗試，以免病從口入	03.wav	4 (方法 5)	3
4	110	公路兩旁的建築式樣單調	02.wav	4 (方法 5)	4
5	114	企業界能突破「同行是冤家」的觀念	06.wav	4 (方法 5)	3
6	120	營建業工人的薪資因地域也有差異	12.wav	4 (方法 5)	3
7	109	二人差點發生肢體衝突	01.wav	4 (方法 5)	4
8	115	倉庫原設計爲放置非危險非易燃物品	07.wav	4 (方法 5)	2
9	112	民眾平日必須步行一公里以上才能取得用水	04.wav	4 (方法 5)	4
10	113	民進黨要求重組中央選委會	05.wav	4 (方法 5)	3
11	116	海洋生態環境的污染越來越嚴重	08.wav	4 (方法 5)	3
12	118	經濟部國貿局今天和相關單位協調	10.wav	4 (方法 5)	4
13	81	開闢山區公路是輕而易舉的事	09.wav	1 (方法 2)	4
14	75	民眾不要冒然嘗試，以免病從口入	03.wav	1 (方法 2)	4
15	80	海洋生態環境的污染越來越嚴重	08.wav	1 (方法 2)	4

圖四十三：評分結果


客語語音合成聽測

主題：客語語音合成聽測

分數統計

	[1] Lo	[2] 1	[3] 2	[4] 3	[5] 4	平均
受測者1	29	47	45	44	39	3.400
平均	2.417	3.917	3.750	3.667	3.250	

標準差

	[1] Lo	[2] 1	[3] 2	[4] 3	[5] 4	總標準差
受測者1	0.493	0.276	0.595	0.471	0.595	0.735
平均	0.493	0.276	0.595	0.471	0.595	

匯出成Excel檔(2007)

匯出成Excel檔(2003)

圖四十四：分數統計表

7.5.4 實驗結果

本論文主要研究內容，是語音合成系統中的文句分析模組。研究項目包括 1.中文斷詞轉客語詞的客語斷詞處理，以及 2.讀音的求取。因此句子念對與否以及是否自然，是本系統主要測試的部分。聽測者聽取合成之語音後，依照表六十九的標準，給予 1 到 5 分的評分。評分標準如下表：

表六十九：平均主觀分數的度量表

分數	品質	失真情形
5	非常好	句子都念對、韻律自然，語音也沒有失真的感覺。
4	好	句子都念對、韻律自然，但語音有點失真，但不覺得煩躁。
3	可	句子有些字念錯、韻律有點不自然，且感覺到失真，而且有點煩躁
2	差	句子有很多字念錯、韻律不自然，聽起來煩躁，但還不至於聽起來不舒服
1	不能接受	句子很多字念錯、韻律不自然，聽起來非常煩躁，而且聽起來不舒服

聽測實驗一：

測試時系統會顯示該客語語音的中文句子，受測者可聽取語音並判斷是否聽得懂，或能猜出意思。在此實驗中，本論文系統輸出的語音有兩種版本：1.在詞內不加入停頓、2.在詞內加入 0.05 秒的停頓，分別合成出 12 句語音檔。加上舊系統合成出的 12 句語音檔，共 36 句。

本聽測實驗，主要想測試對於不熟悉客語的使用者，對於本系統輸出的客語語音之自然度，以及若有中文句子能夠對照時的理解程度。

表七十：自然度及理解程度綜合評分表

	舊系統	**本系統 詞內無停頓**	**本系統 詞內加入 0.05 秒停頓**
Score	3.5	3.05	3.26

由實驗結果顯示，在詞內加入少量的停頓，能幫助聽者理解句子中的意思。因此我們實驗二嘗試加入更長的停頓，並且找來 5 位熟悉客語之人士，測驗文字、文意的正確性與語音的自然度。

聽測實驗二：

測試時系統同樣會顯示該客語語音的中文句子，受測者皆為熟悉客語之人士，因此可判斷句子中的字詞是否念對，也能判斷在聽得懂的情況下，語音是否自然。最後再綜合這些判斷，結果給予 MOS 的評分。在此實驗中，本論文系統輸出的語音有四種版本：1.在詞內不加入停頓、2.在詞內加入 0.05 秒的停頓、3.在詞內加入 0.10 秒的停頓、4.在詞內加入 0.15 秒的停頓，每個版本 12 句，加上舊系統一共 60 句。

表七十一：自然度及文意正確性評分表

	舊系統	**本系統 詞內無 加入停頓**	**本系統 詞內加入 0.05 秒停頓**	**本系統 詞內加入 0.10 秒停頓**	**本系統 詞內加入 0.15 秒停頓**
Score	2.74	3.44	3.43	3.43	3.40

由實驗結果顯示，對於客語理解程度較高的測試者而言，本系統的自然度及文意正確性有明顯的改善。甚至詞內未加入停頓時，有較佳的自然度。最後我們依照實驗一和實驗二綜合的結果，決定在詞間加入少量的靜音，放慢語速讓使用者更容易聽得懂語音內容。

第八章 結論與未來改進方向

本論文針對中文轉客文語音合成系統中的文句分析模組，包含最基礎的客語斷詞語料、國客語對應式語言模型的建置，以及客語斷詞處理、客語讀音求取的方法，已提出一些基礎的研究方法。

針對客語斷詞語料的建置，我們提出了一個有效率的建置方法，並且制定五項標記原則，來儘可能保持標記語料的一致性。經標記時間的統計結果顯示，透過本方法來標記客語斷詞語料，平均每句只需33 秒即可標記出客語斷詞及詞性標記的結果。

針對客語語言模型的設計及訓練，我們提出了國客語對應式語言模型的訓練方法，以「國+客」為單位的訓練方式，可保持國客對應的珍貴資訊。並且透過加成平滑、凱氏平滑以及強化凱氏平滑等語言模型平滑法實驗，找出在語料嚴重資料稀疏的情況下，較適合的解決方案。

針對客語斷詞處理，我們提出了不同於以往架構的方法。我們應用動態規劃演算法及客語 Uni-gram 加 Bi-gram 語言模型的 Mix-Gram 分數算法，於中文詞轉客語詞的處理。實驗結果顯示，系統的 F 分數有 81.41%。相較於傳統中文詞直翻客語詞的方法，已提升不少。

在客語讀音求取上的方法上，我們針對客語發音辭典中的詞目及

讀音，做讀音統計資訊表的訓練。並設計了四段式的流程，來求取客語讀音。實驗結果顯示，本系統讀音求取的正確率有 82.81%。

最後，我們針對文意正確性及自然度，做主觀評分的聽覺測試。實驗結果顯示，本系統合成出的語音已兼具自然度與文意正確性。

目前客語電子語料有嚴重不足的問題，對於本論文所探討的主題而言，是一項非常艱困的挑戰。研究之初，我們並沒有客語斷詞語料可使用，僅有少量一句句未處理的國客語對照文句。因此，我們投入了大量時間在客語語料的標記、建置及辭典的校正、標音。我們也持續的從國小客語教材、客語朗讀比賽文章…等電子文檔中，以人工建置出更多的客語斷詞語料及客語新詞。因此，未來可進行的工作及改進方向有：

1. 擴充國客語對照辭典及客語辭彙發音辭典。
2. 加入客語構詞規則。
3. 探討語言模型平滑化問題。
4. 標記、建置客語語料。
5. 改善語音合成方法，用客語句子訓練出韻律模型、做合成單元。

客語斷詞的應用層面極廣，不僅止使用於語音合成系統中，還可用於客語的數位學習、客語文句處理、客語語音辨識…等系統中。本

論文提出的研究方法，能提供未來客語斷詞相關研究做為基礎與參考。

參考文獻

[1] Algort P. H. and Cover T. M., 1988, A Sandwich Proof of the Shannon-McMillan-Breiman Theorem, Ahe Annals of Probability, Vol. 16, No. 2, pp. 899-909.

[2] Chen Standy F. and Goodman Joshua, 1999, An Empirical Study of Smoothing Techniques for Language Modeling, Computer Speech and Language, Vol. 13, pp. 359-394

[3] Good I. J., 1953, The Population Frequencies of Species and the Estimation of Population Parameters, Biometrika, Vol. 40, pp. 237-264.

[4] H. Jeffreys, Theory of Probability, Clarendon Press, Oxford, Second Edition, 1948.

[5] Jelinek F., Statistical Methods for Speech Recognition, The MIT Press, Cambridge Massachusetts, 1997.

[6] Jurafsky D. and Martin J. H., 2008, Speech and Language Processing (2nd Edition), Prentice Hall, Chapter 6.

[7] Nádas A., 1985, On Turing's Formula for Word Probabilities, IEEE Trans. On Acoustic, Speech and Signal Processing, Vol. ASSP-33, pp. 1414-1416

[8] W. A. Gale and G. Sampson., Good-Turing Frequency Estimation without Tears. Journal of Quantitative Linguistics, 2(3): 15-19, 1995

[9] 江昶毅，應用多種特徵的中文斷詞及詞性標記方法，國立中興大學資訊科學與工程所，碩士論文，2010。

[10] 行政院客委會出版，台灣客家民眾客語使用狀況，2004。

[11] 行政院客委會出版，全國客家人口基礎資料調查研究，2010-2011。

[12] 行政院客委會出版，客語能力認證基本詞彙-中級、中高級暨語料選粹四縣版上冊。

[13] 行政院客委會出版，客語能力認證基本詞彙-中級、中高級暨語料選粹四縣版下冊。

[14] 行政院客委會出版，客語能力認證基本詞彙-初級四縣版。

[15] 吳俊毅，線上客語語音合成系統中產生韻律訊息之研究，國立中興大學資訊科學與工程所，碩士論文，2010。

[16] 呂宜玲，中文語音辨識中語言模型的強化之研究，國立交通大學資訊工程系所，碩士學位論文，2005。

[17] 李雪貞，客語語音合成之初步研究”，國立臺灣科技大學資訊工程所，碩士論文，2002。

[18] 林東毅，客語文句翻語音系統之實作，國立交通大學電信工程所，碩士論文，2007。

[19] 袁里馳，融合語言知識的統計句法分析，中南大學學報自然科學版，2012 年第 43 卷第 3 期。
[20] 袁毓林，基于統計的語言處理模型的局限性，語言文字應用，2004 年 5 月第 2 期。
[21] 張唐瑜，以大量詞彙作為合成單元的中文文轉音系統，國立中興大學資訊科學研究所，碩士論文，2005。
[22] 教育部出版，客家語拼音方案使用手冊，2012。
[23] 教育部出版，教育部客家語書寫推薦用字，2012。
[24] 教育部部編版，客家語分級教材，四縣腔版一到九冊。
[25] 連又箴，台語讀音的求法及標註平臺，國立中興大學資訊科學與工程所，碩士論文，2011。
[26] 陳林、楊丹，獨立于語種的文本分類方法，計算機工程與科學，2008 年第 30 卷第 6 期
[27] 黃健祐，在大量中文語料中語言模型關於平滑問題特性之分析，國立中興大學資訊科學與工程所，碩士論文，2012。
[28] 黃豐隆，國客雙語有聲地圖社群系統，聯合大學資工所，客委會計畫成果報告書，2013。
[29] 黃豐隆，線上國客雙語有聲詞典建置之研究，全國計算機會議 (NCS-2009)，台灣，2009。
[30] 蔡育和，中文文轉音系統中韻律階層的求取，國立中興大學資訊科學與工程所，碩士論文，2005。
[31] 蔡依玲，基於隱藏式馬可夫模型之客語文句轉語音系統，國立交通大學電信程所，碩士論文，2009。
[32] 賴亦傑，應用多詞及多詞性語言模型的中文斷詞及詞性標記方法，國立中興大學資訊科學與工程所，碩士論文，2011。
[33] 鍾屏蘭、江俊龍，學術研究基礎建置暨客家文化研究計畫，屏東教育大學客家文化所，計畫成果報告書，2009。
[34] 羅丞邑，以資料探勘之技術解決線上客語語音合成系統中多音字發音歧義之研究，國立中興大學資訊科學與工程所，碩士論文，2011。
[35] 呂嵩雁，客語《陸豐方言》的百年語言演變析探，國立東華大學台灣語文學系，研究計畫報告書，2007。