The 2019 Conference on Computational Linguistics and Speech Processing
ROCLING 2019, pp. 343-357

國語語音辨識系統中之人名語言模型

The Personal Name Modeling in Mandarin ASR System

梁鴻彬　Hong-Bin Liang
國立交通大學電機工程學系
Department of Electrical Engineering
National Chiao Tung University
hbliang@speech.cm.nctu.edu.tw

王逸如　Yih-Ru Wang
國立交通大學電機工程學系
Department of Electrical Engineering
National Chiao Tung University
yrwang@speech.cm.nctu.edu.tw

摘要

本論文主要有兩個目的：一是訓練一個高效能的中文語音辨識系統；二是改善因人名而造成的 OOV(Out-Of-Vocabulary)問題，並將其辨認出來，以便日後自動轉寫不同類型的語音訊息並產生逐字稿。而人名之辨識對於將來自然語言處理也是一重要的訓練資料。

本論文使用 Kaldi speech recognition toolkit 的環境為基礎，在聲學模型的方面，本實驗使用類神經網路 TDNN 以達到聲音資訊轉成音素序列(phone sequence)的目的；在語言模型方面，本論文透過加入中文特有的語言資訊如形音義詞的合併、專有名詞的拆解，並使用 n-gram 語言模型的訓練，以達到音素序列轉成詞序列(word sequence)的目的，並於解碼過程中調整參數與權重，找出最佳操作點，以得到即時性與辨識率兼顧的語音辨識系統，此外，針對以往人名無法辨認出來的問題，本論文建立特別的人名語言模型以類似 class-based model 的方式置換原 word-based model 中的人名，以達到辨識人名的目的。

Abstract

There are two purposes in the paper, one is training an efficient ASR system, the other is improving the OOV problem caused by the personal name, and we want to recognize it for the purpose of making transcription of different kind of speech data. Name recognition data is also an important training data for the NLP.

The paper base on the environment of Kaldi speech recognition toolkit. In the acoustic model part, we use many different kind of neural network such as TDNN to transform the speech information into phone sequence. In the language part, we add Chinese special

language information such as variant word combination and name entity decomposition, using n-gram language model and lattice rescoring to transform the phone sequence into word sequence. We also tune the parameters and weights during the decoding process to get the best operation point to obtain a ASR system which is not only good at recognition rate but also efficient at recognition time. Moreover, we focus on the problem of difficulty in personal name recognition. We build a class-based like model to replace the original word-based model of personal name to reach the goal of personal name recognition.

關鍵詞：聲學模型、語言模型、中文大辭彙連續語音辨識、時延神經網路(Time Delay Neural Network, TDNN)、專有名詞辨識(Named Entity Recognition, NER)。

Keywords: acoustic model, language model, Mandarin large vocabulary continuous speech recognition, TDNNs, named entity recognition.

一、 緒論

(一)研究動機

隨著訓練語料的增加，電腦運算速度的大幅提升，再加上越來越多製作語音辨識器的工具庫的發明，如 HTK Speech Recognition Toolkit[1], Kaldi ASR[2], et al. 訓練一可用甚至是商用級別的語音辨識已經是唾手可得。

但是，辨識器普遍存在著對於專有名詞辨識率不佳的問題，由台北科技大學廖元甫老師提供之比賽[3]結果，比較國內多所學校與企業所訓練之辨識器，整體辨識率之正確率普遍可高達 90%做右(字錯誤率 10%左右)，但是仔細評估辨認錯誤之詞彙後發現，各辨識器對於專有名詞之辨認率相對較差，尤其在專有名詞中最為重要且為數眾多的人名辨識之正確率僅不到 50%(如圖 一)，也就是說，只要辨識器一遇到人名，辨識率通常不會太高。然而，人名對語言來說又是一個非常重要的資訊，例如在日常生活中，常用的句型裡不外乎由人事時地物所組成，而這裡的「人」，指的就是人名，再者，若辨識器將人名全部當成 OOV 處理，不僅人名這個詞彙辨識不出來以外，也會影響包含人名的句子的辨識能力，因此，本實驗將致力於提升語音辨識器對人名之辨識率，藉由對語言模型之訓練語料加入人名資訊，以期望提高辨識器對於人名之辨識率。

[1] http://htk.eng.cam.ac.uk/
[2] http://kaldi-asr.org/
[3] 指 ISCSLP 2018 Formosa Speech Recognition 會外賽。

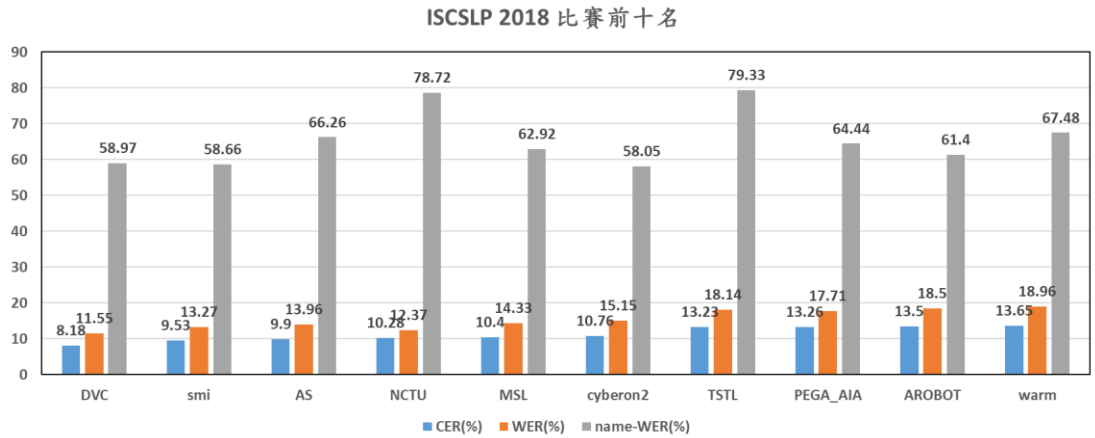

圖 一、比賽前十名之成績（圖例由左至右分別為字錯誤率(CER)、詞錯誤率(WER)與人名詞彙錯誤率(name-WER)）

(二)研究方向

本研究將致力於提升語音辨識器之辨識率與辨識速度，並藉由訓練資料的改善，使之得以辨識以往所不能辨識之人名。

本篇論文首先透過調整聲學模型架構、辭典大小、語言模型參數，使得辨識器在整個語音辨認過程上取得辨識率與辨識速度的平衡。在得到一個辨識率與辨識速度兼具的辨識器之後，將針對辭彙不在辭典裡之問題(Out of Vocabulary, OOV)做改善。

由於人名為 OOV 組成中較為重要、且出現頻率最高的辭彙，本實驗將針對 OOV 裡的人名做特別處理，首先，先建立一 word-based Language Model，並找出所有人名(Word)，將人名拆解成二到三個小單元(Sub word unit)，因本實驗僅要求辨識出之人名的發音正確，部分未被選入辭典之人名將會被轉成音節(Syllable)，以音節的方式加入進辭典裡，之後再對文字語料後處理，將人名資訊加進訓練語料，利用統計特性計算各種不同可能人名互相連接的機率，以產生不同人名之組合，此方法可以讓辨識器對人名的辨識不再侷限於辭典裡出現過的人名，使得語言模型獲得辨識人名的能力。

二、 語料庫介紹

本節將介紹用於本實驗中之所有語料庫。其中用來當作訓練聲學模型之語料的有 TCC300、NER 及 AIShell 語料庫，而為了測試本實驗之辨識系統對於不同環境的辨識能力，因此在測試語料的選擇上，使用 TCC300 及 NER 語料庫，其中 NER 為廣播語料，又可細分為背景乾淨無雜訊之乾淨語料(Clean)以及背景有人為雜訊或音樂參雜其中之其他語料(Other)。

(一)TCC300 語料庫

本次實驗中所使用的 TCC300 麥克風語音資料庫 [1]是由國立交通大學(National Chiao Tung University, NCTU)、國立成功大學(National Cheng Kung University, NCKU)、國立台灣大學(National Taiwan University, NTU)共同錄製而成，並且由中華民國計算語

言學學會(The Association for Computational Linguistics and Chinese Language Processing, ACLCLP)發行，此語料庫屬於麥克風朗讀語音，主要目的為提供台灣腔之中文語音辨認研究使用。其中訓練語料約為 24.4 小時，304780 個音節數；測試語料約為 2.4 小時，26357 個音節數。

(二)NER 語料庫

NER 語料庫 [2]，全名為 NER Manual Transcription Vol1，為國立臺北科技大學和國家教育廣播電台合作錄製之語料庫，主要目的為大量轉寫教育電台之節目，產生節目逐字稿，以建置大規模台灣腔之語料庫，內容大部份為談話性節目，多為自發性(Spontaneous)語音，僅少部分為新聞報導之朗讀式(Reading)語音。其中訓練語料約為 111.5 小時，1715091 個音節數；測試語料可分為乾淨語料(Clean)的 1.9 小時，33660 個音節數與其他語料(Other)的 9.0 個小時，133746 個音節數。

(三)AIShell

AIShell 語料庫 [3]，是由北京希爾貝殼科技有限公司釋放之開源語音資料庫，錄製內容涉及智能居家、無人駕駛等 11 項領域，錄製過程皆在安靜的室內環境。

使用高效能麥克風錄製而成，取樣頻率為 44100 Hz，後降低取樣頻率至 16000 Hz，取樣位元數為 16 位元，由 400 名來自中國不同口音地區的參與者錄製而成，此語料庫文本經人工校正過，正確率為 95%以上。其中訓練語料約為 162.4 小時，1862171 個音節數；測試語料：約為 16.6 小時，178041 個音節數。

三、 深層類神經網路模型配置

由於 CLDNN 網路結構中之 LSTM 層的原因，導致模型之解碼速度大幅的下降，有鑑於此，有學者提出了一種 Resnet-like TDNN-F 模型 [4]，以加深網路層數的方式增強模型之學習能力，且在層與層之間加上一層 Bottleneck layer 降低參數量，以解決參數隨著層數增加而暴增之問題，並加上 Skip connection，跨接上一層所訓練之參數。

此模型亦使用 TCC300、NER、AISHELL 作為訓練語料，特徵參數的抽取為 40 維之 Fbank，並前後串接 2 個音框(2-1-2) ，形成 200 維特徵向量，後經過一個 LDA 矩陣，做為 TDNN-F 之輸入特徵向量，每一層隱藏層神經元個數為 1536，Bottleneck layer 之神經元個數為 160，Bypass-scale 為 0.66，輸出的觀測狀態數目為 2672，總共為 15 層。架構如圖 二。

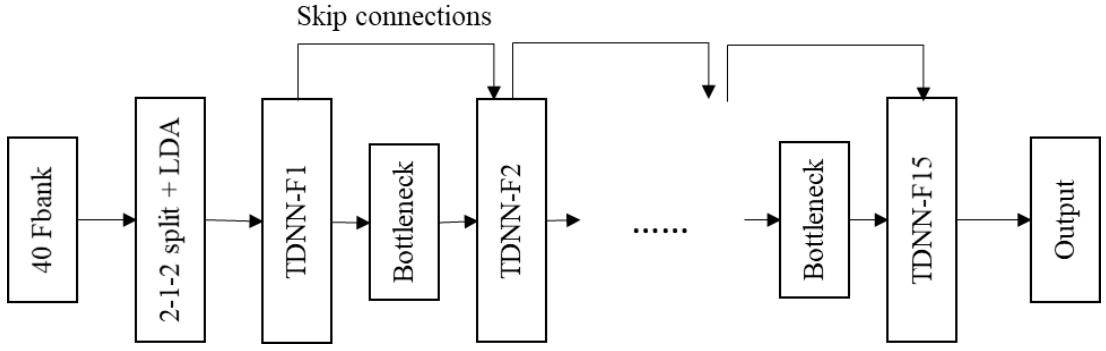

圖 二、TDNN-F 模型架構圖

四、 加入人名語言模型之語音辨識器

在解碼的過程中，若遇到辭典裡沒有出現的詞(Out of Vocabulary, OOV)，則此一辭彙將永遠不可能辨認正確，此時，辨識器會以一相似讀音的詞取代之，使得此一 OOV 附近的辭彙將受到影響，也就是形成所謂的搶詞，進而導致錯誤率的提高[4]。因此，OOV 對於語音辨識來說，一直是個非常重要的問題。在 OOV 組成中，尤以專有名詞(Nb)的出現為最大宗，又專有名詞中，人名為較具意義且出現頻率最高之種類，因此，本實驗將針對人名之辨識率做改善。

為了解決人名被當作 OOV 的問題，本論文將人名從字轉音成音節(G2P)，使原訓練語料中人名的位置是以音節的方式存在，之後再以製造人名隨機填入的方式，使語言模型在原人名位置能看到許多不同種類之人名。詳細內容將在接下來的小節一一解說。

(一)文字語料簡介

本研究用於訓練與研模型之文字語料庫共約 25 億個詞彙，包含以下：

- Chinese Gigaword：由 Linguistic Data Consortium (LDC)整合發行，內容包括台灣中央社、北京新華社等國際新聞。
- 其他由交大語音處理實驗室蒐集之語料

(二)文字語料後處理中之形音義分合詞(variant word)處理

由於訓練語料多來自於新聞語料，為了使訓練資料更接近一般人說話，本實驗將斷詞後的文字做一些特別的處理，使其能更接近口語。例如將文章中出現的科學符號、度量衡等等轉為中文，賦予其統一的文字表示方式，使其被選入辭典後，被語言模型訓練到。後處理中，最重要之步驟如同義詞(variant word)之代換合併。中文之同義詞百百種，能表達同樣意思之詞語的選擇見仁見智，例如同義異音之周一、星期一；同義同音之周一、週一，當人們在使用這些同義詞構句時，這些同義詞之前後連接詞彙幾乎一模一樣，因此，若未合併這些詞彙，統計語言模型將會視這些詞彙為個別不同的詞彙，這將影響辭典收納詞彙之能力，且分散詞句在統計語言模型所統計之機率。依照發音之異同，大致上可以分為兩類，即發音相同與發音相異之詞類，置換原則是建立在字義相同之上，置換的目的一是為了讓文章中同義詞正規化，以利選詞時容納更多詞彙，二是使得合併訓練後的詞可以獲得更多的訓練資料，但是在語言模型建立後，辨識端會產生一個狀況：同義異音之詞彙在合併後將會導致某些發音在解碼圖路徑上消失，導致解碼時無法被搜尋到，如範例中的「禮拜一」被置換成「週一」，故在語言模型中無法找到「禮拜

[4] 這裡指的 OOV 所造成之辨識錯誤率計算方式為 OOV rate 乘以平均詞長。

一」這個詞彙，因此我們在語言模型建置的最後一步，需要處理不同發音之同義異字詞的置換，將「週一」展開成「週一」、「星期一」及「禮拜一」。如表 一。

表 一、同義詞修改範例

形音義分合詞類型	置換前文字	置換後文字
發音相同	周一	週一
發音相異	禮拜一、星期一	週一

本實驗使用實驗室長期累積蒐集之 3160 個同義詞代換，在 7 個不同之語料集上做混淆度(Perplexity, ppl)測試，此實驗使用之語言模型為 4-gram 語言模型，平滑化方法使用 Witten-Bell smoothing[5]，訓練語料為 25 億詞，辭典大小為 120k，差別僅在於一個是使用置換過 variant word 的辭典，另一個是使用未置換 variant word 的辭典。

比較有合併同義詞、與沒有合併同義詞之辭典所建立之語言模型可以發現，有合併同義詞之語言模型的 ppl 較小(見圖 三)，也就是說，平均每搜尋一個詞，所需要使用的辭典數大小較小，表示新的語言模型在詞組搜尋上之效能較佳。由此可見當合併同義詞之後，可以使得語言模型在搜尋詞組上會更有效率。

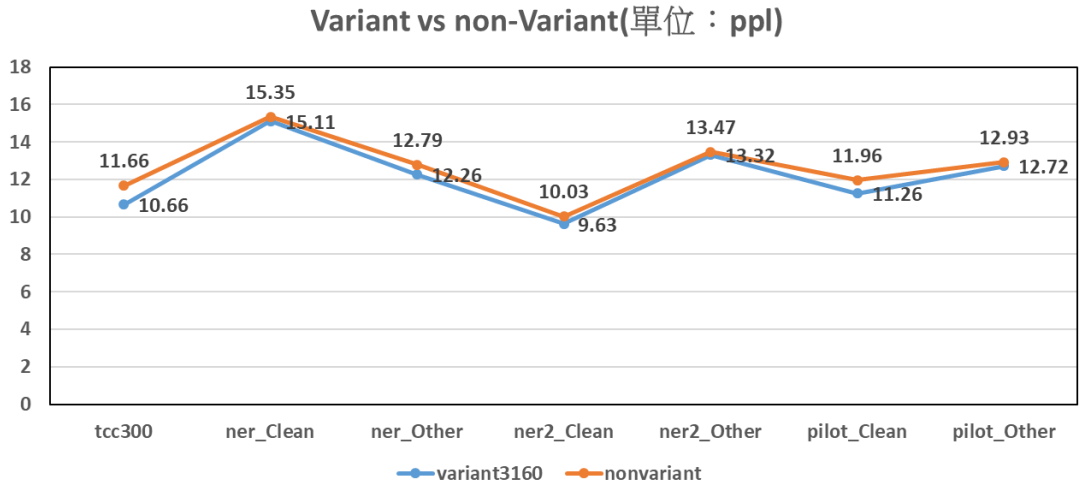

圖 三、辭典使用 120k 且置換 3160 個 variant word 前(non-variant)與置換後(variant3160)之 ppl 比較

[5] 依據經驗，Witten-Bell smoothing 在大詞彙語音辨識系統中的表現會較常見的 kneser-ney 佳。

(三)人名語言模型之建立

語音辨識器之本質是達成聲音轉文字之目的，對於日常對話之句子，將字詞辨認完全正確是非常合理且容易達成的，但是當聽到一人名時，即使是人工轉寫，對於字詞之選擇，仍是無一個百分之百正確之選擇，僅能依日常生活所聽到之常見人名選字選詞，加上中文之同音異字詞實在是多如牛毛，因此，針對辨識器對於人名之辨識結果，僅要求其音節正確，例如「王小明」這個人名將以"wang xiao ming"的形式被辨識器解碼出來。另外，本研究將人名分成兩類：常出現之人名(Somebody)與不常出現之人名(Nobody)，根據本實驗之 25 億詞語料庫統計，人名多為政治人物或新聞上常報導之人物，針對此常出現之人名，取統計數前 5000 名放入辭典中，使其能被以詞的形式解碼出來，剩下之人名，將拆成多個音節放入辭典中，例如「王小明」這一個三字詞將被拆解成"wang xiao ming"三個音節放入辭典中，也就是說，辭典中有關人名的部分將由 5000 個中文人名加上 411 個音節[6]所組成。接下來將介紹本實驗如何建立人名語言模型，使原本被辨識器當成 OOV 之人名能以音節的形式辨識出來。

1. 找出語料中所有人名位置

為了找出文本中出現之人名並置換為#Name，需先將訓練語料中所有人名位置找出來，本研究訂定了一套搜尋規則，首先，利用斷詞器所標記出之詞性(Part of Speech, POS)，挑出所有專有名詞(Nb)，接下來，查詢姓氏排名前三百名[7]作為判斷依據，假如一專有名詞的開頭出現在前三百姓氏表裡，且後續連接字數為一至二字，則判斷為人名，並用#Name 取代之(如圖 四)，以標記此處為一人名，供後續展開人名之處理。

依此規則搜尋在本 25 億訓練語料中，符合人名的個數，總共約為 4100 萬個，約占總訓練語料之 1.6%，字詞之數量僅次於「的」(如下圖 五)。由此可再次看出人名在中文語句中之重要性，不容忽視，若不對人名字詞做專門的處理，僅以一般的辨識器辨認，且假設人名之辨識率為 0%，則所有人名將被當成 OOV 處理，並造成錯一個人名導致前後字詞選詞錯誤之連鎖反應，如表 二，崔蓉芝為人名，此處辨識器因為未在語言模型找到「崔蓉芝」之前後連接詞之機率，而將這一個三字詞解碼為一個一字詞「推」加上一個二字詞「融資」，導致統計模型將計算「推」與「融資」之前後字詞連接機率，造成字詞錯誤接二連三的傳遞下去，也就是形成所謂的搶詞問題。以未經處理之 12 萬詞所建立之語言模型中，統計分析其辨識答案可知，一個二至三字人名平均會造成左右附近 2.03 個詞的錯誤，也就是說，訓練語料中之 1.6%人名最高可造成約 3.25%之詞錯

[6] 中文之所有音節數即為 411 個。
[7] https://news.cnyes.com/news/id/3660142

誤率，這對辨識器來說，實在是一不容小覷之詞錯誤率。

表 二、人名解碼錯誤導致搶詞之問題

正確答案	…	此	案	***	***	崔蓉芝	已經	勝訴	在望	…
辨識答案	…	此	案	推	融資	以及	因	勝訴	在望	…

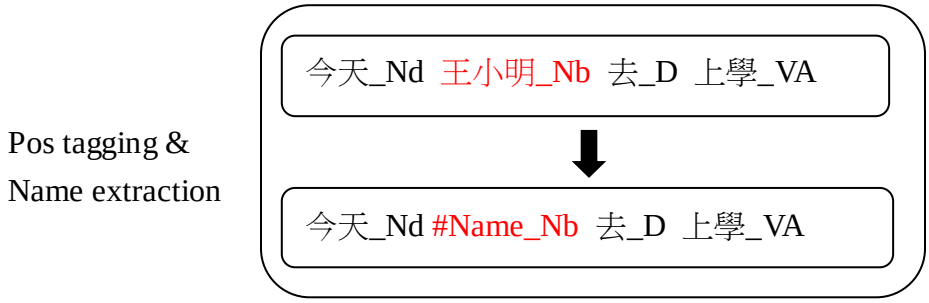

圖 四、找出人名之位置並代換為#Name

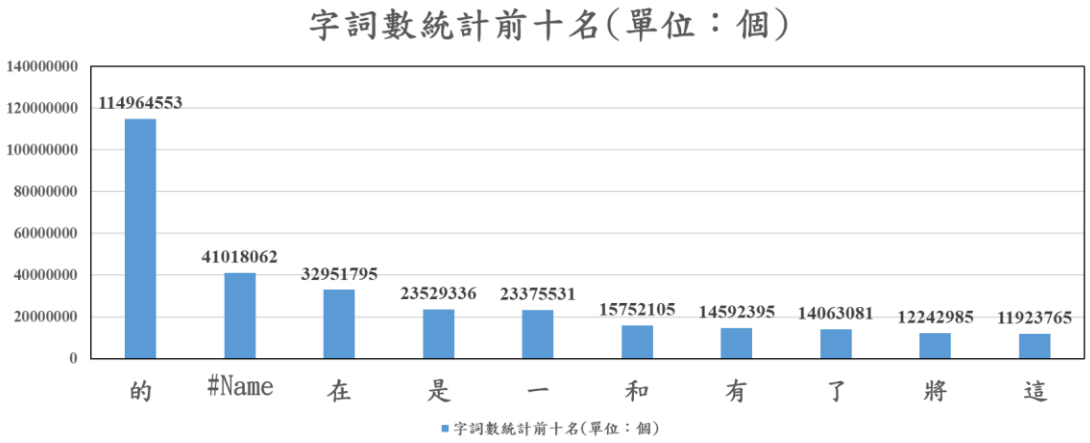

圖 五、字詞數統計前十名

2. class-based 人名語言模型

為了使訓練語料原句中#Name token 處之人名種類更多元，本實驗複製好幾份訓練語料，並根據在前一小節(五之三之 1)所找到的所有#Name token 位置，將所有找到的人名(約 4100 萬個)以隨機亂數的方式填入，若填入之字詞為語料中常出現人名(Somebody)，以字詞(word)填入；若填入之字詞為語料中不常出現之人名(Nobody)，則以音節填入，之後再將這些語料拿去做 LM training。如圖 六。

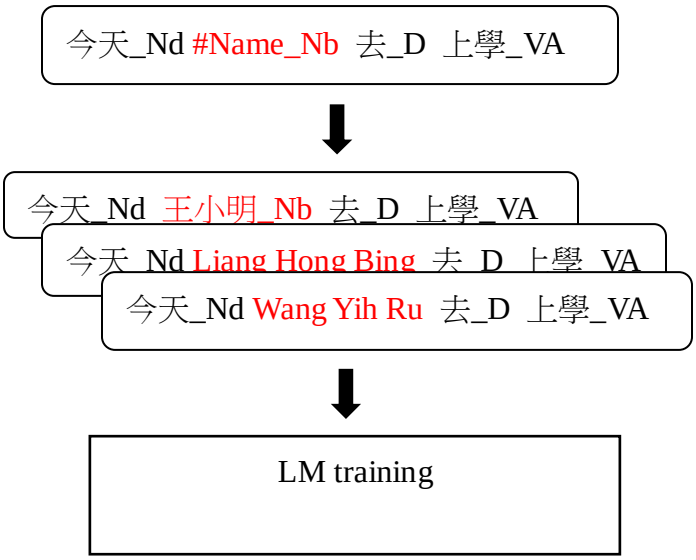

圖 六、製造人名填入#Name 之位置

經由不斷的以亂數方式將人名填入訓練語料中，如此一來，原語言模型中的人名(#Name)位置將出現許多不同的人名，也就是說，語言模型中的人名位置將不再僅能看到原訓練語料中所出現之人名機率，而是類似將#Name 這個標記(Token)展開成一個擁有許多人名機率的 class(如圖 七)，使我們的語言模型組合成 word-based(一般字詞)加 class-based(人名字詞)的語言模型。之後當辨識器遇到一個未曾出現在辭典裡之人名時，將有一定機率被解碼成人名字詞(若為 Somebody)或人名音節(若為 Nobody)。

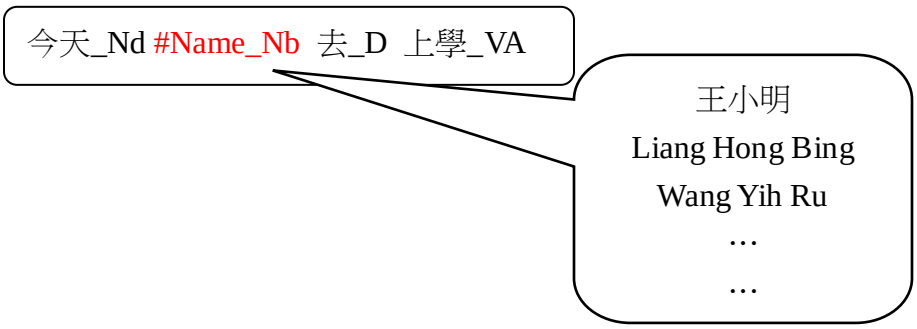

圖 七、人名 word 展開成人名 class

(四)調整一般名詞被當成人名之機率

在之前曾經提到過人名出現之機率非常高(如圖 五)，僅次於「的」，也就是說，辨識器有很高的機率會將一般名詞當成人名來解碼，若剛好欲解碼之詞彙不是人名，且此人名容易與一般名詞混淆時，此將不只影響該位置辨識錯誤，更將影響前後詞彙相連接之機率，例如：有一人名「何平」在訓練語料中出現的機率非常高，當測試句子提到「我喜歡和平」時，此句之意思應是指喜歡和平這種狀態，然而，辨識器因人名之出現機率

過高而將一般名詞「和平」解碼為人名「何平」，進而解碼成「我喜歡何平」，表示喜歡一個叫做何平的人，此將導致不僅「和平」這個字詞辨識錯誤，該位置之前後詞彙連接機率也會大不相同，將有可能會發生搶詞的情況。

因此，為了降低此種錯誤發生，本實驗額外訓練一完全不包含人名之語言模型，並定義一權重 α，以內插的方式(interpolation)與人名語言模型合併，試著降低詞彙被當人名之機率，如圖 八，詳細之實驗結果將在第六章做更詳細的說明。

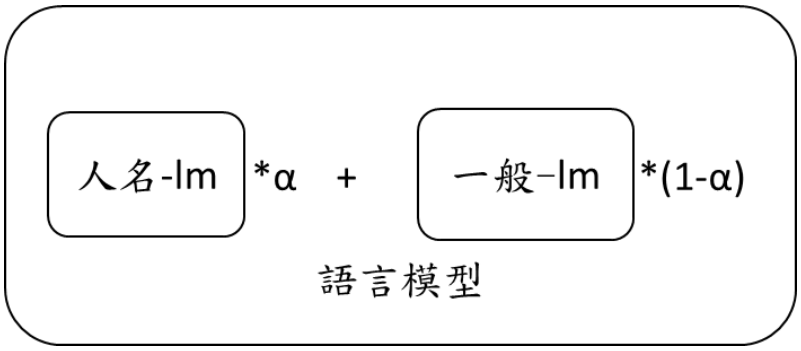

圖 八、降低人名語言模型之權重後之語言模型。

五、 實驗結果分析與討論

此實驗之測試語料除了 tcc300、ner_Clean、ner_Other 外，另加入同樣為自發性語音、新聞廣播語料之 ner2_Clean(音節數 12171 個，約 52 分鐘)、ner2_Other(音節數 223385，約 12.6 小時)、pilot_Clean(音節數 63695，約 4.3 小時)、pilot_Other(音節數 107326，約 6.8 小時)。

在評估辨識率之前，本章節將先說明因人名而導致之搶詞情形的嚴重性，並介紹錯誤率之計算方式，最後，再以調整語言模型之內插權重的方式，控制各種錯誤的發生機率。

在前面的章節曾經提到過，若放任人名在辨識器中解碼，則人名將很有可能被拆解成由許多小字數詞組(Sub-word unit)所組成，而這些 sub-word unit 又將影響前後字詞之連接機率，導致錯誤傳遞下去，例如先前提到的例子「崔蓉芝」，但也有可能很幸運地，辨識器以存在於辭典之發音相似之人名取代之，在此情況下，錯誤僅發生在人名本身，並不會發生搶詞的情況，綜合上述兩種情形，定義人名之錯誤傳遞率(Error Propagation Rate, EPR)，計算方式如式(4.1)。由於本實驗之人名語言模型會將人名解碼為音節或文字，在計算 EPR 時仍會先將辨識正確之音節記為錯誤，直到後面再做校正。本測試語料之總詞數為 387510 個，總人名為 1812 個，在加入人名語言模型後，比較各測試語料之人名錯誤傳遞率，即每單位人名所造成左右詞彙之錯誤量，如表 三。在加入人名語言模型後，僅僅在錯誤傳遞率上，就由原先的 2.03(未加入人名語言模型前)降至 1.37，也就是說，原先人名 OOV 為 1%將造成 2.03%WER 的結果，將降為 1.37%。

$$EPR = \frac{\text{人名所導致之錯誤個數}}{\text{人名錯誤個數}} \tag{4.1}$$

表 三、加入人名語言模型前後之 EPR 比較

	加入人名語言模型前	加入人名語言模型後
tcc300	1.84	1.13
ner_Clean	2.25	1.52
ner_Other	2.03	1.37
ner2_Clean	1.78	1.23
ner2_Other	2.41	1.66
pilot_Clean	1.73	1.17
pilot_Other	2.38	1.71
Overall	2.03	1.37

在評估人名語言模型之優劣時，與一般詞彙不同，且在評估時，本實驗僅針對人名解碼成音節之正確率做觀察，換句話說，那些本來就存在於詞典裡的有名人物(Somebody)將自然而然的解碼成字詞，不是此實驗之重點。在加入人名語言模型後，雖然理想上可增加原本被當成 OOV 的人名被辨識出來的機率，但同時，也可能造成幾種因人名語言模型的加入所導致的錯誤，因此，本實驗定義了兩種類型的錯誤，與一種類型的正確方式，說明如下：

1. False Alarm(type I error):

一般字詞被當成人名解碼。這裡指的一般字詞也包含 OOV，即所有非人名字詞。此類錯誤是由於語言模型中人名詞彙之出現機率過高，導致辨識器選用人名來替換一般詞彙。例：「長話短說」被解碼成「張華頓 說」。

2. Wrong Detection(type II error):

人名被當成一般字詞解碼。此類錯誤是由於人名詞彙與一般常用詞彙之發音相似，導致辨識器解碼錯誤。例：「何平」被解碼成「和平」;「鍾國仁」被解碼成「中國人」。

3. Hit:

人名被當成人名解碼。不論解碼出的人名音節是否完全正確，只要該詞彙在辨識器中是被當成人名來看待，就算 Hit。

為了評估辨識器在解碼人名詞彙之能力，定義下列參數：

定義辨識器抓到之所有人名且為 Hit 的比率 Precision 為：

$$\text{Precision} = \frac{\text{Hit}}{\text{Hit+False Alarm}} \tag{4.2}$$

定義測試語料中所有人名且為 Hit 的比率 Recall 為：

$$\text{Recall} = \frac{\text{Hit}}{\text{Hit+Wrong Detection}} \tag{4.3}$$

以調和平均數綜合以上兩個參數，計算其 F1-score：

$$F1score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \tag{4.4}$$

當 Precision 越低，表示 False Alarm 越多，非人名被當成人名的情況越嚴重，反之亦然；當 Recall 越低，表示 Wrong Detection 越多，人名沒有被當成人名的情況越嚴重，反之亦然。

各測試語料上之結果如下圖 九所示：

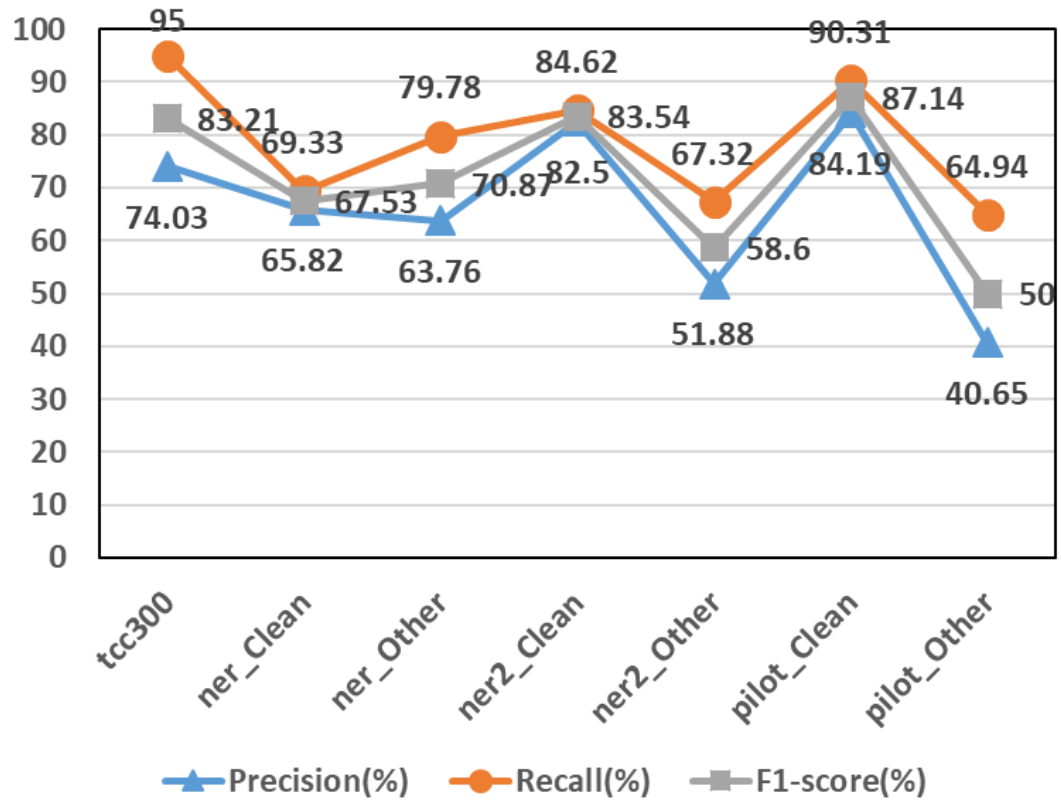

圖 九、人名辨識能力之評估

由實驗結果得知，各測試語料之 Precision 普遍偏低，這是因為辨識器容易將 OOV 當成人名來辨識，而此類錯誤又被計入 Precision 之緣故，表 四為各測試語料，OOV 佔 False Alarm 的比例。

表 四、False Alarm 中為 OOV 的比例

Tcc300	Ner_Clean	Ner_Other	Ner2_Clean	Ner2_Other	Pilot_Clean	Pilot_Other
76.67%	29.63%	53.01%	78.57%	67.19%	79.22%	54.34%

欲控制辨識器將詞彙當成人名字詞辨識的機率，以調整 Flase Alarm 或 Wrong Detection 的高低，本實驗將訓練語料中不含人名之句子特別分離出來製做一個完全不含人名之語言模型，並給定一個權重將此語言模型內插進原本的人名語言模型中，以稀釋辨識器整體人名詞彙出現之機率如式(4.5)。 分別調整 α 為 1.0(人名語言模型完全未

調整前，即圖 九使用之人名語言模型)、0.5、0.1，結果如圖 十所示。當人名語言模型權重越來越低時，Precision 因 OOV 越來越不容易被當成人名辨識而提高；Recall 將因人名越來越不容易被當成人名辨識而降低。

$$\mathrm{LM} = \alpha \mathrm{LM}_{\text{人名}} + (1-\alpha)\mathrm{LM}_{\text{不含人名}} \tag{4.5}$$

α=1.0

	Tcc300	Ner_Clean	Ner_Other	Ner2_Clean	Ner2_Other	Pilot_Clean	Pilot_Other
Precision(%)	74.03	65.82	63.76	82.50	51.88	84.19	40.65
Recall(%)	95.00	69.33	79.78	84.62	67.32	90.31	64.94
F1-score(%)	83.21	67.53	70.87	83.54	58.60	87.14	50.00

α=0.5

	Tcc300	Ner_Clean	Ner_Other	Ner2_Clean	Ner2_Other	Pilot_Clean	Pilot_Other
Precision(%)	75.57	72.86	68.25	81.82	51.33	83.99	47.54
Recall(%)	92.78	68.00	78.26	80.77	65.85	88.99	63.04
F1-score(%)	83.29	70.34	72.91	81.29	57.69	86.42	54.21

α=0.1

	Tcc300	Ner_Clean	Ner_Other	Ner2_Clean	Ner2_Other	Pilot_Clean	Pilot_Other
Precision(%)	86.13	85.45	81.58	85.29	61.62	89.14	66.32
Recall(%)	82.78	62.67	66.67	74.36	55.61	79.52	55.17
F1-score(%)	84.42	72.31	73.37	79.45	58.46	84.05	60.24

圖 十、調整詞彙被當成人名之機率

前述實驗在計算 Hit 的時候，僅要求解碼出的答案為人名，而 Hit 裡又可再細分為三種類型：1.音節完全辨識正確，例如「梁鴻彬」解碼為「liang_hong_bin」。2.音節相似但不完全正確，例如「郭振興」原本應解碼為「guo_zheng_xing」但辨識器解碼為「guo_zheng_xin」，一個是「ㄒㄧㄥ」的拼音，一個是「ㄒㄧㄣ」的拼音，由於人們在平常講話時「ㄥ」跟「ㄣ」本來就分辨不出來，故此種音節相似之解碼錯誤實在是無可厚非。3.由一個發音相似，且存在於詞典裡之人名代替之，例如「黃朝興」被辨識器解碼為「黃昭星」，其中，「黃昭星」為有被選入詞典中之常出現人名，也就是前面章節所定義的 Somebody。

在所有測試語料之總詞彙數 387510 個、總人名詞數為 1812 個中[8]，本實驗之辨識器能辨識人名的正確率(hit/總人名數)約為 81%；辨識出人名且音節完全正確[9]之正確率(hit 且音節完全正確/總人名數)約為 73%，也就是說，即使測試語料出現一個從未出現在詞典中之人名，本辨識器仍有 73%之機率將其音節完全正確的辨識出來。

最後，為了觀察人名語言模型對於整體辨識率的影響，我們重新計算加入人名語言模型後辨識器之詞錯誤率，這裡僅將 Hit 中音節完全解碼正確之人名當作解碼正確，並與未加入人名語言模型前之辨識器做比較，實驗結果如下圖 十一所示。此實驗使用之

[8] 這裡的 1812 個人名包含存在於詞典之人名與未存在於詞典之 OOV 人名。
[9] 這裡指僅計入 Hit 中音節完全正確之類型。

聲學模型為上述(第四章)的 TDNN-F 架構，語言模型為 4-gram 語言模型，平滑化方法使用 Witten-Bell smoothing，訓練語料為 25 億詞，辭典大小為 120k，唯一差別僅在於語言模型有或沒有加入人名語言模型做調整。

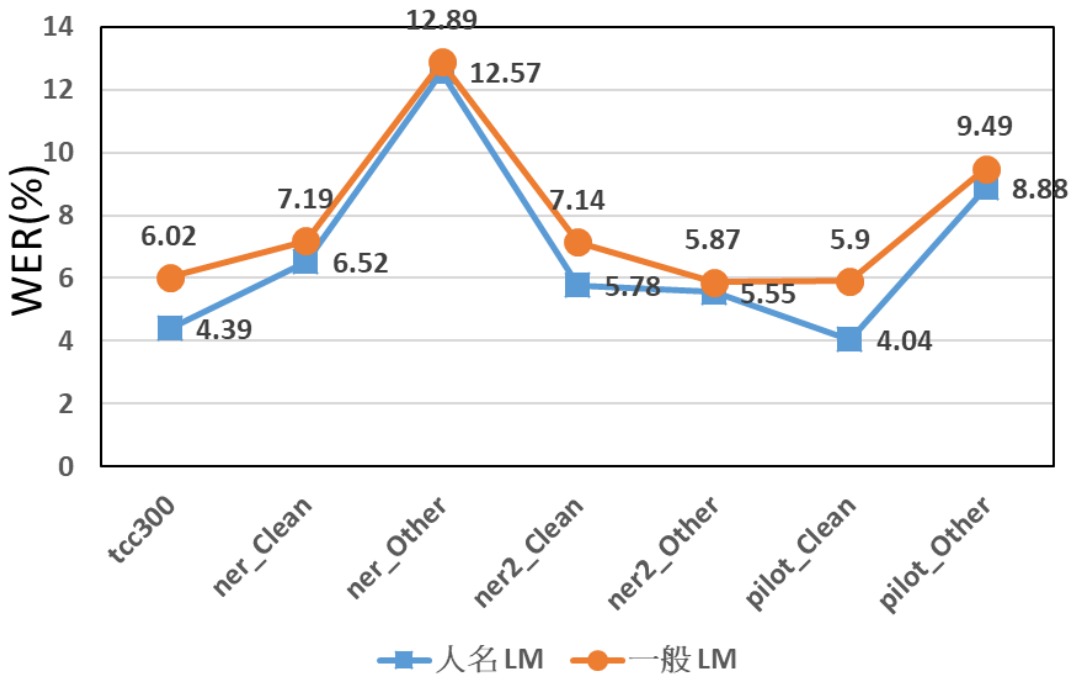

圖 十一、加入與未加入人名語言模型之詞錯誤率比較

六、 結論

參考 Kaldi 之作者 Daniel Povey 在 2018 發表之論文，使用 TDNN-F 之架構下訓練之聲學模型，辨識速度的確比傳統之 LSTM 快很多，即使音節辨識率有些差距，那些差距在後級語言模型解碼後也顯得微乎其微。

在人名辨識率方面，加入人名語言模型可以使以前被當成 OOV 處理之人名被解碼出來，不但救回了以前不在詞典裡之人名以外，也減少了左右搶詞的情形。

本實驗建構之大詞彙中文語音辨識器統對於乾淨語料(Clean)有絕佳的辨識能力，但是對於噪音環境(Other)語料之辨識率仍然有改進之空間，尤其在聲學模型方面，本實驗並沒有對噪音做特別的處理，導致在噪音環境下之音節錯誤率仍然偏高，這會使得即使後級之語言模型再強，某些字詞還是無法挽救回來之情形，而語言模型如果能克服記憶體限制之問題往 5-gram 語言模型發展，將會使辨識率又再更為提升，尤其在加入人名語言模型後，訓練語料裡之人名被展開至 2 到 3 個音節數，使用 5-gram 語言模型將可以看得更遠，以增加人名詞彙之 Hit 數。此外，目前實驗室辨識器對於未出現在詞典裡之人名是以音節的形式解碼出來，也許未來能以機率或其他搜尋的方式，選擇大家普遍能接受的字詞呈現。

參考文獻

[1] "Mandarin Microphone Speech Corpus-TCC300," [Online]. Available: http://www.aclclp.org.tw/use_mat_c.php#tcc300edu.

[2] "Formosa Speech Recognition Challenge 2018," [Online]. Available: https://sites.google.com/speech.ntut.edu.tw/fsw/home/challenge#h.p_I_b8URx26NXZ.

[3] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, “AIShell-1: An open-source Mandarin speech corpus and a speech recognition baseline,” Proc. Oriental COCOSDA, 2017.

[4] Daniel Povey, Gaofeng Cheng, Yiming Wang, Ke Li, Hainan Xu, Mahsa Yarmohamadi, Sanjeev Khudanpur, "Semi-OrthogonalLow-RankMatrixFactorizationforDeepNeuralNetworks," in *interspeech*, 2018.