The 2017 Conference on Computational Linguistics and Speech Processing
ROCLING 2017, pp. 213-229

完全基於類神經網路之語音合成系統初步研究

A Preliminary Study on Fully Neural Network-based Speech Synthesis System

廖書漢 SHU-HAN Liao[a] ,蔡亞伯 Ya- Bo Chai[a] ，廖元甫 [a] Yuan-Fu Liao,

[a] 國立台北科技大學電子工程系

sam8105111@gmail.com, d0030253@gmail.com, yfliao@ntut.edu.tw

摘要

傳統的語音合成使用先文字分析後語音合成的架構，但是這種兩階段的作法，通常會有，若前級分析錯誤，就會影響後級合成，且無法挽救的問題。因此，在本論文中我們希望嘗試把前後級，全部都改成以類神經網路實現，以便將來可以直接合成一個大的端對端語音合成類神經網路。主要的想法是，直接以字元串為輸入單位，並盡量用大量未標記語料，進行非監督式類神經網路訓練。我們的系統包含四個子網路，分別是DNN_G以sequence-to-sequence[1] [2]架構作字轉音，DNN_C以word2vec[3]擷取characterclass ， DNN_T以recurrent neural networklanguage model（RNNLM）[4]，求取字元時序關係，與DNN_S以deep neural network進行語音合成。實驗語料由專業播音員錄製，內容包括孟德爾傳全書以及從網路擷取約3000句的中英夾雜句子。並以相同文字要求新舊系統各自合成測試語料，請10人進行聽測試，分別以新舊系統各聽10句，進行A/B/X偏好度測試，與以新舊系統各聽20句，做mean opinion score（MOS）評分，評估新舊系統的可理解度，自然度與相似度。從實驗結果發現，在可理解度、自然度和相似度方面，分別有72%、70%和61%的人偏好新系統。而且新系統的可理解度、自然度和相似度的MOS主觀分數各為3.59、3.1和3.18分，高於舊系統的3.33、3.03和2.9分，顯示我們所提出的系統效能相當不錯，印證我們提出的想法確實可行。

關鍵詞：**語音合成、深度類神經網路、端對端**

一、簡介

傳統語音合成系統中，包含兩個處理階段，分別是前端文本分析與後端聲音合成(如下圖1)。其中在前端文本分析模組，包含了文字正規化、斷詞、字轉音、詞

性(part of speech，POS)標註[5]等文字分析，以求取文脈訊息。另一方面在後端的聲音合成模組，則透過前級求出的的文脈訊息特徵參數，進行語言合成模型訓練，以合成聲音訊息。

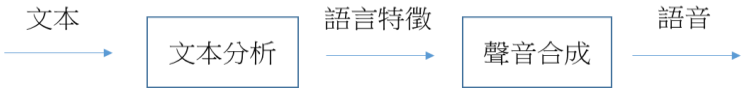


圖 1 傳統 TTS 系統架構圖

但在這個架構下，前級和後級都是獨立建構的，因此，如果前級出現問題的話，後級也會只能繼承前級的錯誤，但卻無法往回調整，以更正前級的錯誤。而且，傳統上，前級所用parser，都是使用自然語音專家發展好的現成系統，並不容易自行修改。

因此，我們希望能把前後級，全部都改成以類神經網路實現，以便將來可以直接合成一個如圖2的端對端語音合成類神經網路，避免傳統兩階段架構的缺點。

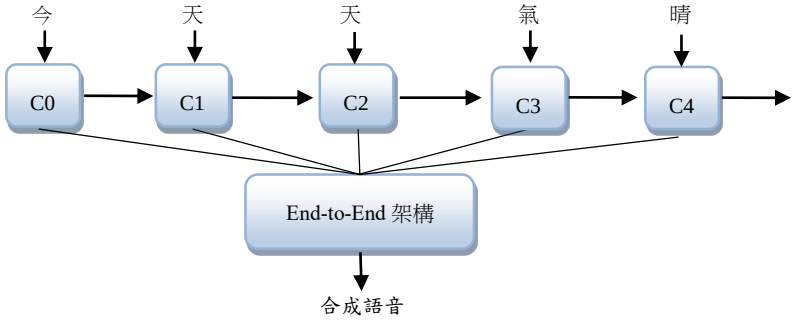


圖2 字元層級的End-to-End架構

為達到此目的，我們完全用類神經網路，取代前級的字轉音，文字分析，與後級的語音合成模組。主要的想法是，先將原本的系統改成直接以字元串為處理單位，再使用四個子網路，分別是DNN_G以sequence-to-sequence架構作字轉音，DNN_C以Word2vec擷取每個輸入字元的類別與文法屬性，DNN_T以RNNLM，求取每個字元在整個句子中的狀態跟時序關係，與DNN_S以deep neural network接收各個網路的隱藏層神經元激發資訊，進行語音合成。

在此架構中的DNN_G因為是使用sequence-to-sequence模型，可以處理未曾看過的字詞的發音，DNN_C與DNN_T所使用的Word2vec與RNNLM更可以善用大量隨手可得

的未標記文字語料，進行非監督式訓練，充分訓練整個類神經網路，避開傳統parser，需要依賴人工標記語料，才能進行才能進行訓練的問題。

二、朝向端對端語音合成架構

為了朝向以基於字元層級的端對端語音合成系統，我們將整個端對端語音合成系統，以圖3的方式，切分成4塊DNN_s。分別是文字轉拼音網路DNN_G、字元屬性與角色分類DNN_c、字元時序關DNN_T和語音合成DNN_s。

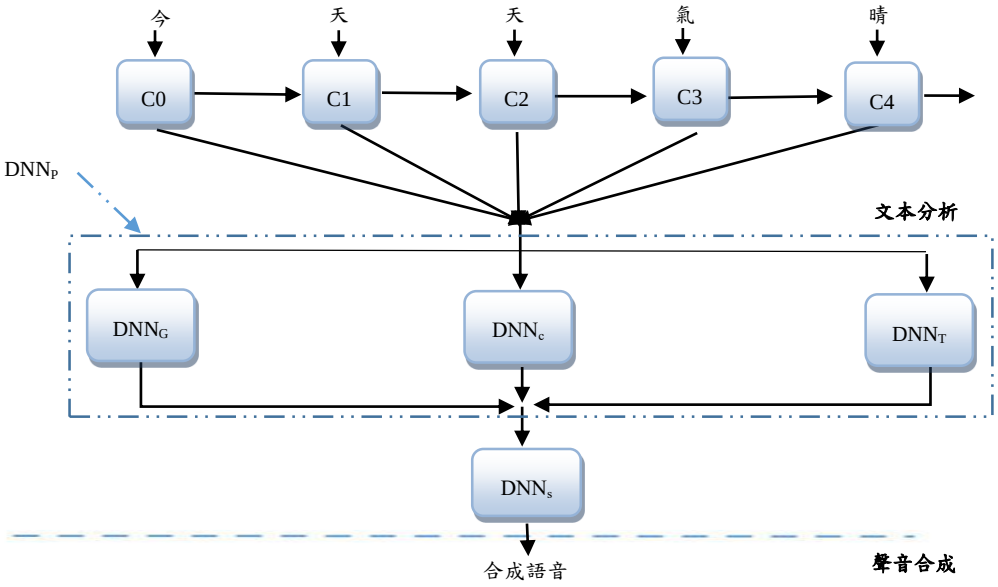


圖3 One-Stage架構內部功能方塊

其中字轉音(DNN_G)主要是利用sequence-to-sequence模型轉換文本的拼音和音調，DNN_c使用 Word2vec 來抓取字元特性，而DNN_T 透過RNNLM來擷取字元前後時序資訊關係。最後形成一個所有的文脈訊息，都是由類神經網路自動產生。這樣一來便能避開傳統文本分析的諸多不便。最後後級聲音合成部分則使用DNN_s來接收DNN_G、DNN_c與DNN_T擷取出的文脈訊息，以合成語音。以下進一步詳細敘述各子網路的實際作法。

(一)、字轉音(DNN_G)

Seq2Seq 全名是 Sequence to Sequence，Seq2Seq 就像一個翻譯模型，比如輸入序列是英文(hello)，輸出序列是中文(你好)，該技術改善了傳統輸入序列和輸出序列長度需要一樣的問題，開始了將深度神經網路模型(DNN)運用在機器翻譯這類

型的任務。Seq2Seq 最早是由兩篇文章闡述他的主要思想，分別是 Google 的 Sequence to Sequence Learning with Neural Networks[1]和 Yoshua Bengio 團隊的 Learning Phrase Representation using RNN Encoder-Decoder for Statistical Machine Translation[2]，這兩篇文章針對機器翻譯的問題不約而同的提出相似的解決想法，Seq2Seq 由此產生。

在字轉音方面利用Seq2Seq技術，Seq2Seq全名是Sequence to Sequence，Seq2Seq的核心想法就是透過深度神經網路模型(常用的是Long-Short Term Memory，LSTM)，將一個輸入的序列映射到一個輸出的序列。而這過程包含兩個環節，分別是將輸入編碼和解碼產生輸出。在這個模型中每一個時間的輸入和輸出是不一樣的，例如現在的輸入編碼序列是「上班族EOS」，其中EOS(End of Sentence)為句尾識別符號，依序將「上」、「班」、「族」、「EOS」傳入模型中，將輸入序列映射為解碼輸出序列「ss_ch-A:_ch-N_chp_ch-A:_ch-n_chts_ch-u:_ch<EOS>」。

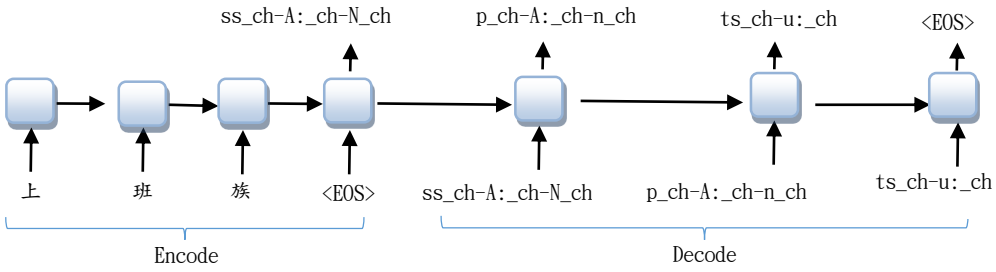


圖4 Google的Sequence to Sequence架構[1]

此外為使Seq2Seq的G2P架構可以考慮到前後文的內容，進而給予當前字一個較為可能的發音，以處理多音字的問題，因此我們一次不是只輸入一個字，而是同時包含其前後關係。

表1為進行Seq2Seq的G2P訓練時的輸入資料形式[] []，主要模仿CNN開一個sliding window去掃前後的字，讓它能往前與往後多看5個字，以獲得更多訊息，而能學得更好。

表1 G2P訓練資料格式

句子/單字	拼音
各行各業皆顯著的改善不	xian3
行各業皆顯著的改善不少	zhu4
各業皆顯著的改善不少,	de0
○○○○○隔閡○○○○	ge2
○○○○隔閡○○○○○	he2
○○○○○三○○○○○	san1

(二)、字元語意與文法屬性(DNNc)

word2vec[3]能夠將輸入的字詞轉換到向量空間進行計算，分析後可以在向量空間中發現，相聚在一起的向量轉換回文字後，會是相近屬性的詞彙。word2vec能夠將字詞語意和文法角色做分類，而且它不需要給標註過的文字語料就能訓練，這可以避開傳統 parser 需要人工標註的繁雜工作，也能夠做到類似 POS 的功能。

我們利用Word2Vec，訓練如圖5的類神經網路，將字元轉到向量空間。訓練完成後，再擷取隱藏層神經元的word vector輸出向量，當作每個字元的語意與文法角色資訊。主要是將大量未標註語料倒入Word2vec，讓他自行訓練，再利用求出之字元向量，界定每個字元的屬性與文法角色關系。

也因為Word2vec所訓練出來的字向量空間，能有意義的表示字的屬性，並且能夠將訓練出來的字向量進行排列，讓類似屬性的字聚類在一起，所以我們覺得用它來替代傳統Parser中的標註詞性功能可能是行得通的。

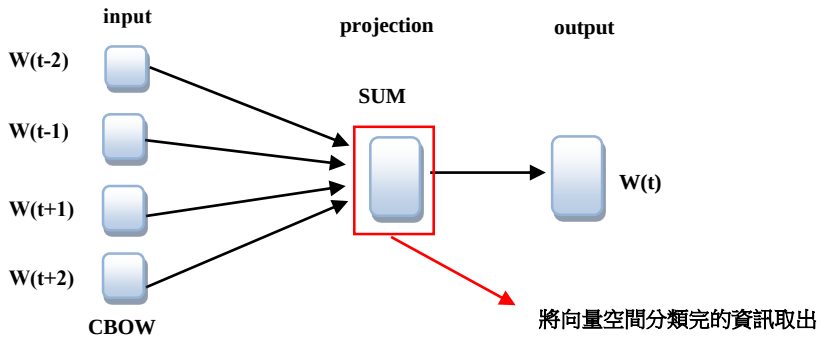


圖 5 擷取隱藏層分類資訊

(三)、字元時序狀態(DNN_T)

在我們的架構下，我們希望能不使用 parser 就從文本獲得每個字元在當前語句的狀態，讓機器自動學習文章的脈絡，進而能從目前語句預測到下一句可能為何。遞迴神經網路在 Distributed Representations of Words and Phrases and their Compositionality [4]此篇論文也指出使用遞迴神經網路模型進行訓練能從隱藏層中的連續輸出向量獲得字詞在語句中的狀態。

本文所使用的是Mikolov的RNNLM，不過我們是使用字元階層來進行訓練。RNNLM能夠直接使用無標註文章語料，進行訓練，並因其擁有記憶能力，能夠學習到較長時間的文章脈絡。所以我們用大量無標記語料，訓練完RNNLM後，藉由擷取RNNLM隱藏層神經元的激發狀態值（如圖6所示），當作某一字元在文章段落中的時序狀態資訊。此外，我們並進一步進行量化，整理成0與1的值，用來表示某一字元在句子中的時序狀態資訊。

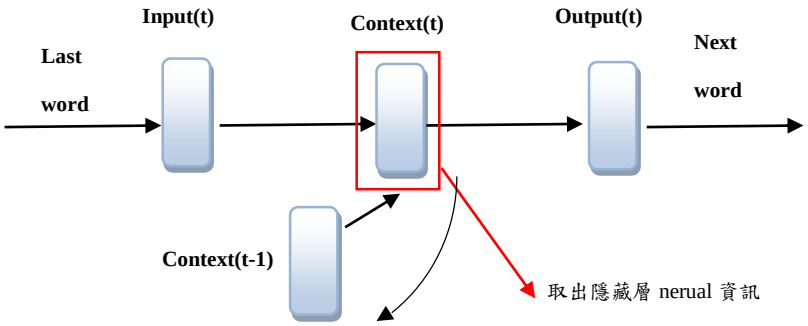


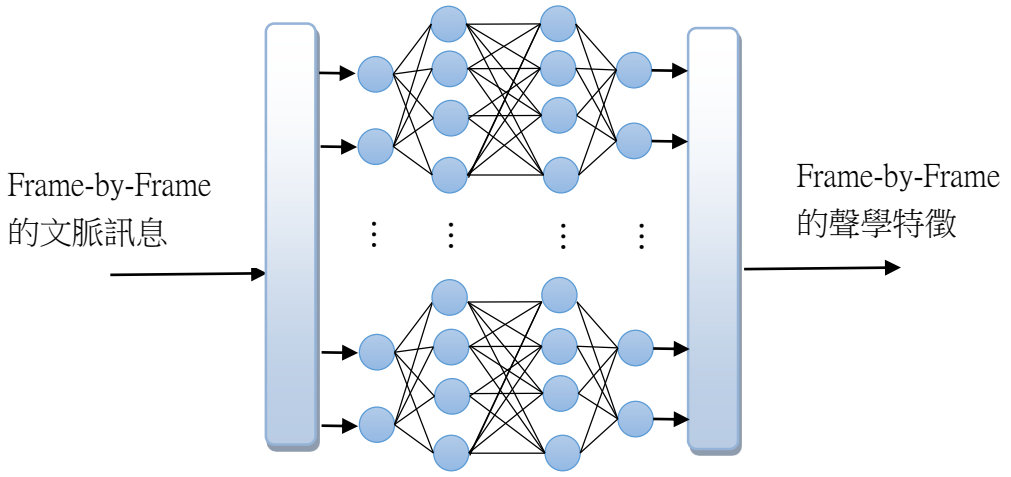
圖6 擷取隱藏層時序資訊

四、DNN 架構(DNN_S)

在本論文中，為了將整個架構用成 One-Stage，後級聲音合成勢必也要替換成類神經網路的架構，在我們的 One-Stage 中利用了 HTS[6] version 2.3.1 中新添加的 frame-by-frame modeling option using DNN based on HMM state alignment，來接收前面 3 個 DNN 所萃取出的資訊並且訓練合成語音。

前面3個DNN所萃取出的資訊，可以進一步合併成為完整的文脈訊息。DNN_S要學習的就是DNN_P輸出的文脈訊息，與訓練語料間的對應關係。在DNN_S這邊，

會先從聲音語料中擷取Frame-by-Frame的聲音特徵參數，先前3個DNN所擷取的資訊也會做成Frame-by-Frame的格式，再以如圖7所示的架構，進行和成模型訓練。



三、語音合成實驗結果與分析

為了與傳統方法作比較，舊系統使用Parser做文本分析和查表字轉音，而新系統則是使用DNN_P、DNN_C與DNN_T來分析文本，DNN_S來合成語音。新舊系統皆使用相同的中英夾雜語料訓練。我們要比較的是新舊系統合成音檔的相似度、可理解度、自然度的偏好度與MOS分數。為求公平比較，選用沒有在訓練過程中出現的語料來合成，最後比較的音檔皆為同樣文字，但受測者不知道那個音檔為新或舊系統所合成，以盲測方式進行

(一)、實驗設定

1、字轉音語料

在DNN_G部分，G2P訓練語料分別為TCC300與我們實驗室有的10萬字詞字典，TCC300是文章性質的語料，而10萬字字典則是以單字詞和多字詞為主，如表2。

表2 G2P訓練語料庫

	字數	性質
TCC300 文字語料庫	286685	文章
10 萬字詞字典語料	231477	單字、詞

2、文本語料

在DNN_C與DNN_T部分，我們使用Mikolov團隊的open source，分別是word2vec與RNNLM Toolkit，訓練這兩個模型的語料為Chinese Gigaword Second Edition + Wikipedia。Gigaword與Wikipedia語料的統計資料如表3所示。

表3 word2vec與RNNLM訓練語料

	Chinese Gigaword Second Edition	wikipedia
語料性質	主題式文章	名詞解釋
句子總數	共約 1200 萬句	

3、語音合成語料

本實驗使用的語音合成訓練語料，是我們與台灣數位有聲書協會合作錄製的"NTUT Audiobook Corpus Vol.2"。合成的測試語料則是從中抽取中文100句及中英夾雜100句來作合成，抽出的句子皆不在訓練語料之中。表4為訓練語料資料表，表5則為合成語料的資料。

表4 訓練語料資料表

	中文語料	中英夾雜語料	英文語料
語料內容出處	生命科學大師：遺傳學之父	線上文本	CMU
語料句數	約 4800 句	約 3500 句	約 990 句
每句詞數	20-35 詞	10-30 詞	5-15 個單字
時間長度	約 170 分鐘	約 200 分鐘	約 79 分鐘

表5 合成語料資料表

	中文語料	中英夾雜語料
語料內容出處	生命科學大師：遺傳學之父孟德	線上文本
測試語料音檔數	2 個	2 個
測試語料總句數	192 句	100 句
測試語料每句字數	依文章為準	10-30 字

表6 HTS的DNN參數

Number of Hidden&Units	3layers1024units
activation	Sigmoid
optimizer	Adam
Batch size	256
learnRate	0.001

4、文脈訊息求取方法與設定

新方法與舊方法中整個前級文本分析完全不同，舊文脈訊息依然採用Parser來進行文本分析和查表字轉音；新系統則是將舊系統求取文脈訊息的方式都去除，字轉音部分採用DNN_G轉換字詞拼音語調，原本Parser部分則是使用DNN_C與DNN_T來分析，並利用DNN_T的隱藏層狀態來代表字元在語句中的時序關係。

(二)、評估方法

評估測試包括合成音檔可理解度，相似度與自然度的偏好度與MOS主觀分數，我們將測試音檔給10位母語為國語的人士進行評分，新舊系統偏好度測試是2選1的方式，為標準的A/B/X測試，新舊系統請每人各聽10句；而平均主觀值分數請每人各聽20句，評分方式為1~5分，分數越高則為越好。

(三)、聲音合成實驗結果

1、前級文本分析實驗結果

a.G2P

為了評估我們DNN_G使用Seq2Seq的G2P能否替換掉原本的字轉音方式，我們以中英夾雜文字語料(工研院提供之線上文本)、大陸文章(Blizzard Challenge 2010的測試語料)與擷取於國家文學博士/國立師大教授許錟輝主編的「常見破音字」一書，和<薛意梅>常用的100個破音字中的破音字集來做比較，看不同情況下新舊G2P各自的正確率為何。實驗結果如表7、表8與表9所示。

表7　中英夾雜文字語料新舊G2P的正確率

一般文章	總共 450 句(約 7330 字)	Accuracy
舊 G2P	錯 26 字	99.64%
新 G2P	錯 32 字	99.56%

表8 大陸文章中新舊G2P的音節正確率

一般文章	總共 5880 句(約 31 萬個音節)	Accuracy
舊 G2P	錯 3096 個音節	99.0%
新 G2P	錯 8488 個音節	97.3%

表9 破音字句子中新舊G2P的破音字正確率

破音字句子	總共 227 句(7988 字,破音字 509 字)	破音字 Accuracy
舊 G2P	錯 65 字	87.22%
新 G2P	錯 58 字	88.6%

由表7、8、9可以看見在一般文章中，新G2P與舊G2P差距不大，但是在大陸文章中，新G2P的表現並沒有很好；而在破音字測試中，新G2P稍微好些，總之新的G2P效能與原本舊系統的效果相當，未來可再多添加破音字訓練語料，才進一步改善正確率。

b. Word2vec實驗

在DNN_C與DNN_T這部分我們分別做各自的實驗，從實驗結果來看到底有沒有符合我們的預期，表10為利用Word2vec分析Chinese Gigaword Second Edition + Wikipedia文字語料庫，將字元轉成向量後，再以字元向量作分群的結果。由此可知將字元轉換到向量空間處理分析，可以有效的將字元語意、字元文法角色等屬性擷取出來。

表10 character embedding 產生的字分類

分類	字元
1	弊、斂、案、涉、瀆、疑、蒐、貪、賂、賄
2	墾、壩、岩、岸、島、峽、崗、嶺、嶼、巒
3	絮、絹、綢、綴、綿、緞、緻、縷、繡、繪
4	い、が、と、き、り、で、る、く、し、も
5	個、分、呎、哩、尺、斤、釐、秒、年、頃
6	吧、呀、呢、哦、啊、啦、喔、嗎、嗯、呵

c.RNNLM 實驗

在表 11 可以看見利用 Chinese Gigaword Second Edition + Wikipedia 文字語料庫訓練出來的 RNNLM，在利用 RNNLM 模型產生出的句子中，語意或是時間順序上都貼近新聞文字。因此應用 RNNLM 能學到時序關係的特性，剛好與傳統文本分析中的時間順序關係相似，所以將其應用在我們的想法上應該是可行的。

表11 使用RNNLM產生之文句

時序方向	範例
Forward	在民主黨的表現,他們1致認為,1切不可能會對我們的壓力.
Backward	.遇待國惠最的國美開離定決已,判談的國美與國合聯在國美兼理總副國美,示表統總李

2、替換文字分析模組，對語言合成的影響

語音合成實驗的部分我們透過替換，比較各元件，分為以下三部分，包括(1)舊G2P與新G2P,(2)ParserPOS資訊與Word2vec,(3)Parser斷詞時序位置與RNNLM輸出之隱藏層狀態，來做整體語音合成評估。

a、替換舊G2P與新G2P語音

在此先單獨探討單純的G2P元件，使用Seq2Seq的G2P會不會與傳統的G2P有很大的差異性，以及替換後的聲音合不合理，在此我們使用少量合成音檔測試一下兩者的偏好度與MOS分數。

實驗結果顯示在圖8與表12中，我們可以看見單單只有換掉G2P元件的時候，新舊系統兩者其實相差不會很多。

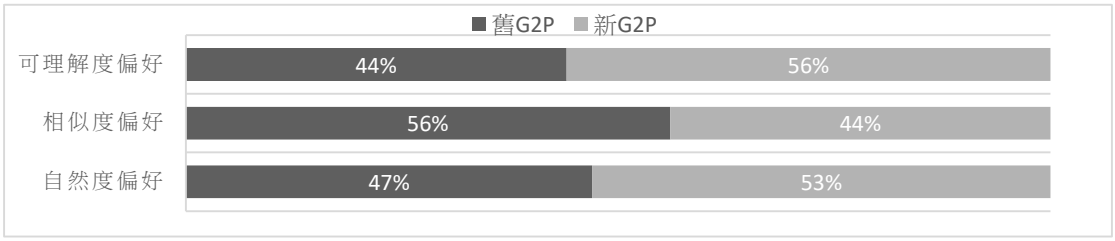


圖8 新舊G2P偏好度比較

表12 新舊G2P之MOS主觀分數比較

純中文	舊 G2P	新 G2P
可理解度評分	3.1	3.3
相似度評分	3.1	3
自然度評分	2.9	3.1

b、替換Parser詞及POS資訊與Word2vec影響之比較

本實驗利用Word2vec來求取字元的語意和文法角色資訊。在此我們與傳統Parser的POS資訊來做比較，觀察字元的語意文法資訊的取代與傳統方式的差異。

在比較新舊系統的POS資訊時，因為G2P還是必需要有的，所以在比較上可能會受到新舊G2P和新舊文法腳色資訊的互相拉扯，不過由圖9與表13的實驗結果來看，總體來說還是改用Word2vec的資訊後的新架構比傳統使用Parser的舊架構稍好一些。

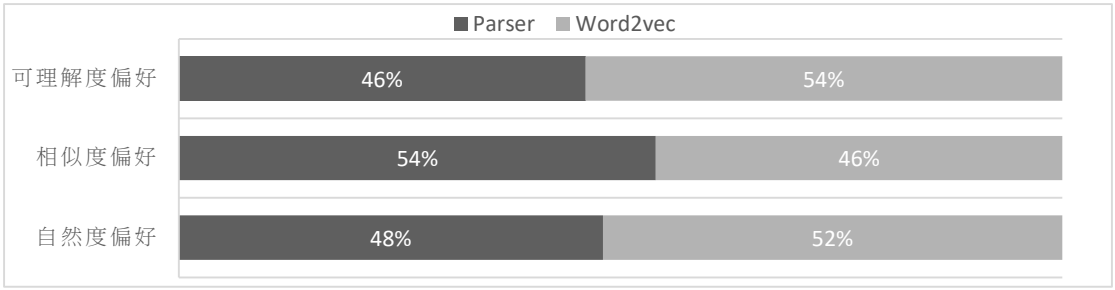


圖9 新舊架構之語意文法偏好度比較

表13 新舊文法角色之MOS主觀分數

純中文	Parser	Word2vec
可理解度評分	3	3.2
相似度評分	3.1	3
自然度評分	3	3

c、替換Parser時序位置資訊與RNNLM影響之比較

時序關係的實驗結果比較顯示在圖10與表14中。總體來說也是改用RNNLM的資訊的新架構比傳統使用Parser的舊架構稍好一些。

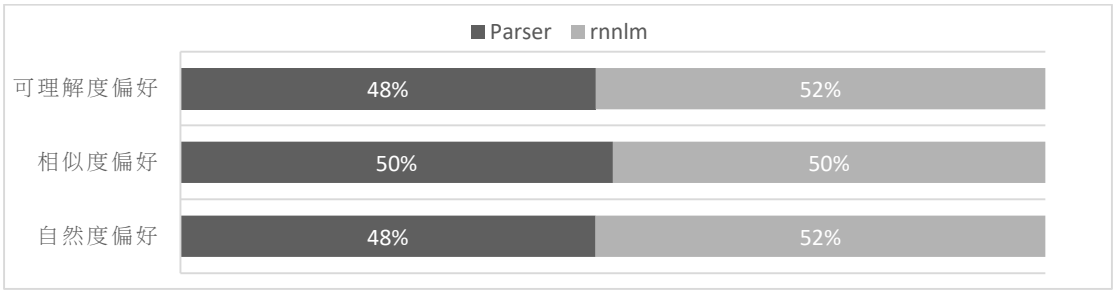


圖10 新舊架構之時序關係偏好度比較

表14 新舊時序關係之MOS主觀分數

純中文	Parser	RNNLM
可理解度評分	3.1	3.3
相似度評分	3.2	3
自然度評分	3	3.1

3、整體新舊系統架構聲音合成偏好度比較

因為我們還缺少 duration model，還不能直接算出每個聲音要合成多長。所以目前是先用 HTS 的 duraiton model 估算合成長度。再加上擷取前級三個網路的輸出當文脈資訊，建立後級要用的 frame-by-frame 合成資訊。主要是前級跟後級如何連接，如何添加 label 中的文脈資訊。

在聲音合成實驗結果部分，我們採用偏好評比以及平均主觀值分數，分別測試以下四種架構，探討前後級的不同組合會有甚麼影響：

- Parser[7][8]+HTS = 前級Parser，後級HTS(傳統Two-Stage語音合成系統)。
- Parser+DNN$_S$= 前級Parser，後級HTS的DNN。
- DNN$_P$+HTS = 前級DNN$_G$ + DNN$_C$ + DNN$_T$，後級HTS。
- DNN$_P$+DNN$_S$= 前級DNN$_G$+ DNN$_C$ + DNN$_T$，後級HTS的DNN(End-to-End)。

a、DNN vs HMM語音合成偏好度比較

此部份我們以比較Parser+HTS與Parser+DNN$_S$架構跟比較DNN$_P$+HTS與DNN$_P$+DNN$_S$架構比較，來看以傳統HMM和以DNN作語音合成的差異性。

實驗結果如圖11和12所示。從圖11和12可以看出在說話自然度上以DNN合成的聲音會稍微自然一些，特別是在英文上，但差異不大。

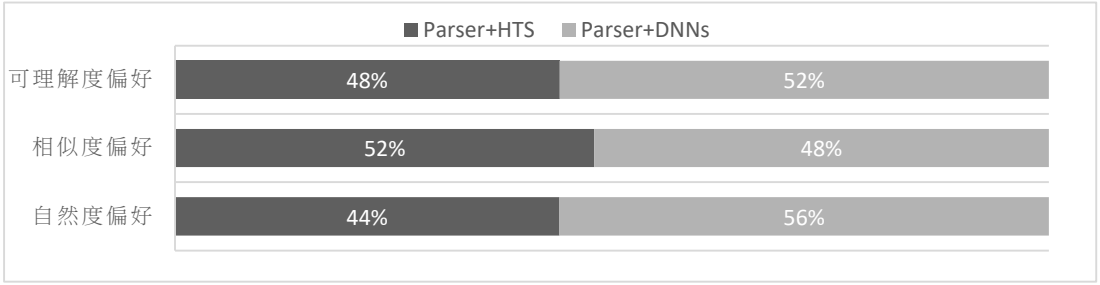


圖 11 Parser+HTS 架構與 Parser+DNN$_S$

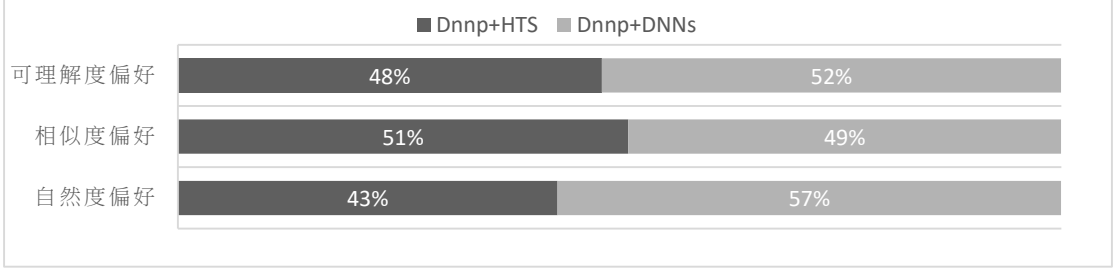


圖 12 Dnn$_P$+HTS 架構與 Dnn$_P$+DNN$_S$ 架構偏好度比較

b、Parser vs DNN$_P$語音合成偏好度比較

在文本分析方式的偏好比較中，我們比較Parser+HTS與DNN$_P$+HTS架構，跟比較Parser+DNN$_S$與DNN$_P$+DNN$_S$架構，來了解以Parser方式和DNN$_P$方式求取文脈對語言合成聲音的影響差異。

圖13和14為實驗結果，可以看出文本分析對合成語音的影響。實驗結果顯示，使用DNN$_P$的架構不管是對HMM或是DNN$_S$都會比較好，而且有相當差距。顯然將前級換成DNN$_P$對語音合成影響相當明顯。

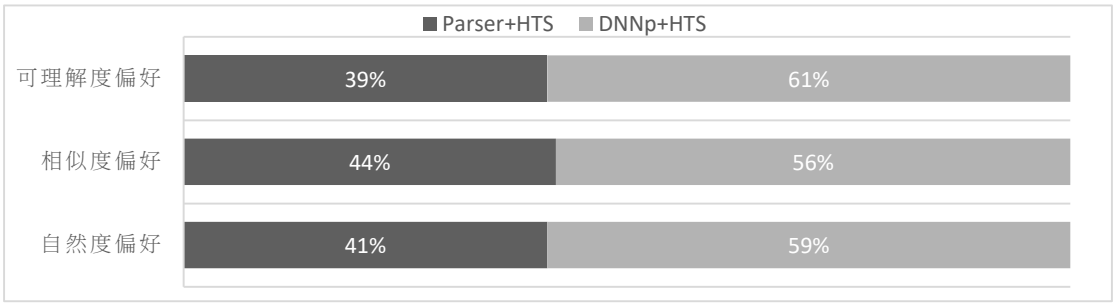


圖13 Parser+hts與DNN$_P$+HTS架構偏好度比較

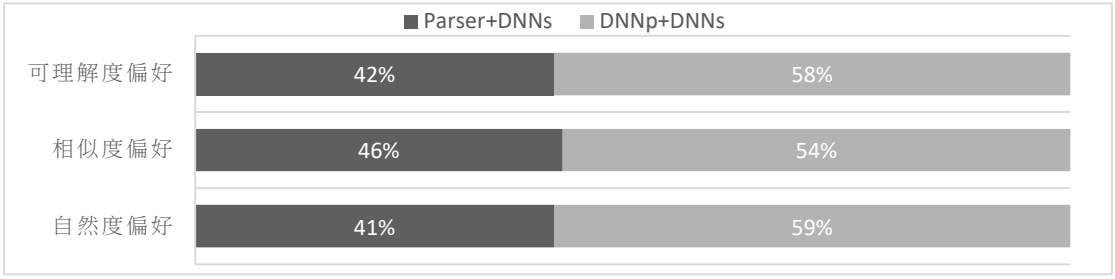


圖14 Parser+DNN$_S$與DNN$_P$+DNN$_S$架構偏好度比較

c、Parser+HTS與DNN$_p$＋DNN$_s$語音合成偏好度比較

整體新舊系統的效能比較為本論文的重點，在此我們進一步比較Parser+HTS架構(傳統Two-Stage語音合成系統)與DNN$_P$+DNN$_S$號架構(新的DNN架構，前級換成DNN$_p$、DNN$_c$與DNN$_T$，後級使用DNN$_S$)，來看我們所提出的新架構有沒有比傳統的Two-Stage還要好。實驗結果如下圖15所示。顯然使用DNN$_p$+DNN$_s$架構，所合成的聲音明顯好很多。

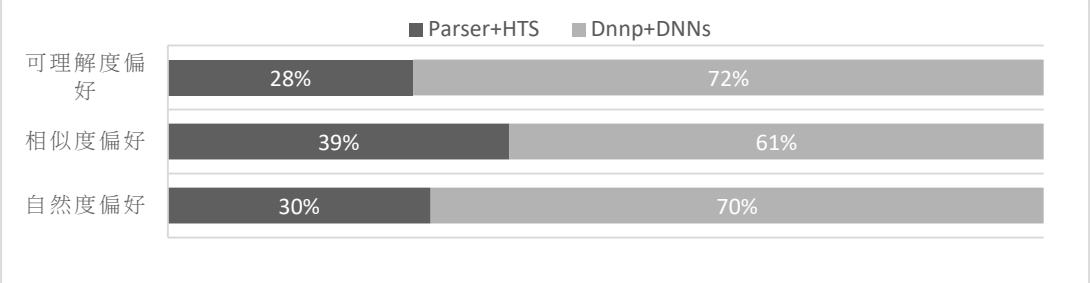


圖15 Parser+HTS架構與DNN_P+DNN_S架構偏好度比較

d、四種架構的MOS主觀分數比較

表14與15則分別為（1）Parser+HTS架構、（2）Parser+DNNs架構、（3）DNNp+HTS架構與（4）DNNp+DNNs架構，在純中文及中英夾雜語音合成的MOS分數，由表13和14來看DNNp+DNNs架構，所合成的聲音確實比較好。

表14 四種架構的中文聲音MOS主觀分數比較

純中文	Parser+HTS	Parser+DNN_S	DNN_P+HTS	DNN_P+DNN_S
可理解度評分	3.38	3.47	3.6	3.56
相似度評分	3.01	3.1	3.05	3.1
自然度評分	2.95	3.14	3.16	3.13

表15 四種架構構中英夾雜聲音MOS主觀分數比較

中英夾雜	Parser+HTS	Parser+DNN_S	DNN_P+HTS	DNN_P+DNN_S
可理解度評分	3.28	3.3	3.56	3.62
相似度評分	3.05	3.21	2.93	3.1
自然度評分	2.85	3.05	3.1	3.23

五、結論

在本論文中，我們將傳統前級文本分析拆成 DNN_G、DNN_c 與 DNN_T 三個部分，後級則使用 DNN_s 來做語音合成，這是初步嘗試，以後會建一個大網路，包含所有子網路，並以目前的子網路的訓練結果當大網路的係數的初始值。最後直接量測輸出合成語音的錯誤成本函數，回過頭來訓練整個系統。實驗結果總結在表 16 與圖 16。結果顯示新系統所合成的聲音在各方面都勝過於傳統 Paser+HTS 的語音合成系統。因此以各方面來看，整個架構使用神經網路來實做的確會比

較優越，這也證明了朝向 End-to-End 的語音合成架構的這個想法是可行的。

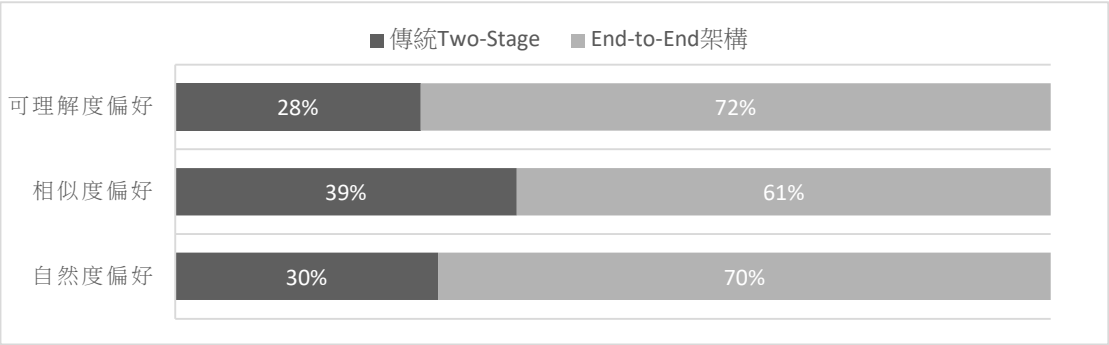


圖16 傳統Two-Stage與新系統架構的架構的偏好度比較

表16 傳統Two-Stage與End-to-End架構的MOS主觀分數比較

	傳統 Two-Stage 架構	End-to-End 架構
可理解度評分	3.33	3.59
相似度評分	3.03	3.1
自然度評分	2.9	3.18

參考文獻

[1] Ilya Sutskever, Oriol Vinyals, Quoc V. Le. Sequence to Sequence Learning with Neural Networks, In arXiv:1409.3215v3 [cs.CL] 14 Dec 2014

[2] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, Yoshua Bengio. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In arXiv:1406.1078v3 [cs.CL] 3 Sep 2014.

[3] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. Advances in Neural Information Processing Systems 26 (NIPS 2013).

[4] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality, Advances in Neural Information Processing Systems 26 (NIPS 2013).

[5] Eric Brill, A SIMPLE RULE-BASED PART OF SPEECH TAGGER ,1992.

[6] HMM-based Speech Synthesis System (HTS) : http://hts.sp.nitech.ac.jp , 2016, July.

[7] Yih-Ru Wang, Yuan-Fu Liao, Yeh-Kuang Wu, Liang-Chun Chang,“Conditional Random Field-based Parser and Language Model for Traditional Chinese Spelling Checker” Proceedings of the Seventh SIGHAN Workshop on Chinese Language Processing (SIGHAN-7), pages 69–73,Nagoya, Japan, 14 October 2013.

[8]Stanford-Parser : http://nlp.stanford.edu/software/lex-parser.shtml , 2016, July.
[9] Sequence-to-Sequence G2P toolkit : https://github.com/cmusphinx/g2p-seq2seq
[10] Jason Lee, Kyunghyun Cho, Thomas Hofmann. Fully Character-Level Neural Machine Translation without Explicit Segmentation. In arXiv:1610.03017v1 [cs.CL] 10 Oct 2016.