The 2019 Conference on Computational Linguistics and Speech Processing
ROCLING 2019, pp. 182-191

基於卷積神經網路之台語關鍵詞辨識

劉祈宏　Chi-Hung Liu

長庚大學資訊工程學系

Department of Computer Science and Information Engineering
Chang Gung University
m0729015@cgu.edu.tw

呂仁園　Ren-Yuan Lyu

長庚大學資訊工程學系

Department of Computer Science and Information Engineering
Chang Gung University
renyuan.lyu@gmail.com

詹惟中　Wei-Zhong Zhan

b0429016@cgu.edu.tw

吳捷書　Jie-Shu Wu

b0429031@cgu.edu.tw

朱達道　Da-Dao Zhu

b0429052@cgu.edu.tw

施俊良　Jun-Liang Shi

b0429063@cgu.edu.tw

長庚大學資訊工程學系

Department of Computer Science and Information Engineering
Chang Gung University

摘要

本文使用近年興起的卷積神經網路模型來對於台語特定關鍵詞進行訓練，訓練準確度可達 9 成，我們使用 TensorFlow speech command 所使用的 30 單字翻譯成台語，其中包含了十個數字(零到九)及其他日常生活中常用的單詞並提供使用者透過我們建立的平台做單詞音檔即時辨識。

一、 緒論

語言流失已經是全球各地最普遍的危機之一。以台灣為例，目前語言溝通上幾乎以國語為主，幾乎都是老一輩的人在使用台語溝通。根據統計，幾乎大部分會講的年輕人都是七八年級生，也提到家庭環境非常重要。如果從小跟家中長輩經常使用台語，溝通上就不會有太大問題。因此為了保存及推廣台語文化，我們希望開發語音辨識系統，目前網路上並無免費開源台語音檔。首先，我們希望為台語辨識研究者建立一個能夠收集音檔的平台，並針對收集到的音檔進行單詞語音辨識的開發，並提供使用者透過我們建立的平台做單詞音檔即時辨識。

二、 相關研究

由於近年來 GPU 與深度學習與人工智慧興起，我們採用深度學習中的卷積神經網路，主要架構採用此篇論文 [1]，而資料收集方面參考 [2]。在市面上，目前開源的語音資料庫如 LibriSpeech[1], Mozilla Common Voice[2], timit[3] 等皆有國語或英語等語料，但唯獨台語資料在市面上是非常缺乏的，而台語的學習成本也是較高的，臺灣話與其他漢語系語言同為聲調語言，聲調在語句中有辨義作用，亦有不少繁複的變調規則以及文白讀異。

[1] http://www.openslr.org/resources.php 中文語料 SLR18, SLR38, SLR47, SLR62
[2] https://voice.mozilla.org/zh-TW/datasets 華語(台灣) 語料
[3] https://scidm.nchc.org.tw/dataset/darpa-timit DARPA TIMIT 英文語料

三、 台語簡介

台語（英文：Taiwanese Hokkien、Taiwanese，又稱為臺灣閩南語。近代以來常以台語（臺羅：Tâi-gí/gú/gír）稱之。以其為母語的閩南裔臺灣人是臺灣第一大族群。

台灣從明朝末年開始，陸續有部分中國大陸沿海居民遷徙至台灣，特別是”清領後期”渡臺禁令開放後，大量福建南部（閩南）的泉州府和漳州府的居民紛紛遷徙至台灣。由於台灣先後分別歷經了荷蘭及西班牙的統治，後有明鄭與清朝統治，1895 年後更由日本統治長達 50 年，造成閩南語逐漸在台灣各地演變分化，並融入荷蘭語、日語及原住民語言等語言於其中，使得台語與福建的閩南語在詞彙使用及腔調上存在有不少差異，台語也逐漸成為臺灣本島最主要的通行語言之一，而根據 2009 年所發表的《臺灣年鑑》中指出，臺灣民眾約有 73%能夠說台語。

總體上說，台語在北部為偏泉混合腔，中南部平原偏內埔腔，西部沿海偏海口腔。漳州移民主要居住在中部平原地帶、北部沿海地區及蘭陽平原，被稱為內埔腔；泉州移民主要居住在中部沿海地區、臺北盆地，被稱為海口腔，南部則為泉漳混合區。

四、 TensorFlow

TensorFlow 是 Google 基於 DistBelief 進行研發的開源機器學習框架。第一，在這裡會採用 tensorflow 作為開發的原因主要為開源，這意味著世上有志之士皆可以是參與者，提報臭蟲，優化程式碼，使整個社群可以是朝氣蓬勃的。第二為 Google 公司的維護，Google 在 2018 年的市值約 7255 億美元，有如此財力的公司作為後盾，可以確保此框架的延續性。第三，TensorFlow 支援的程式語言種類繁多，目前主流的機器學習框架如 Caffe、PyTorch、CNTK 大多支援 2~3 種語言，或者只有一種，而 TensorFlow 共支援了 python、C++、R、Swift、JavaScript、Go 等… ，可以讓程式設計師更靈活地面對到各種情境，靈活部署。第四、針對行動裝置或是物聯網裝置，TensorFlow Lite 讓模型可在多樣裝置上執行，包括行動裝置、物聯網裝置……等等，意即可以在 Raspberry Pi 或您的手機上，進行機器學習，讓應用場景多樣化。 第五、範例程式碼

與完整的說明文件，我們都知道萬事起頭難，有了官方所提供的範例程式，對於新手而言可以更快地上手，清晰地說明文件可以省下查找程式碼用法時間，提高效率。

五、 台語語音辨識

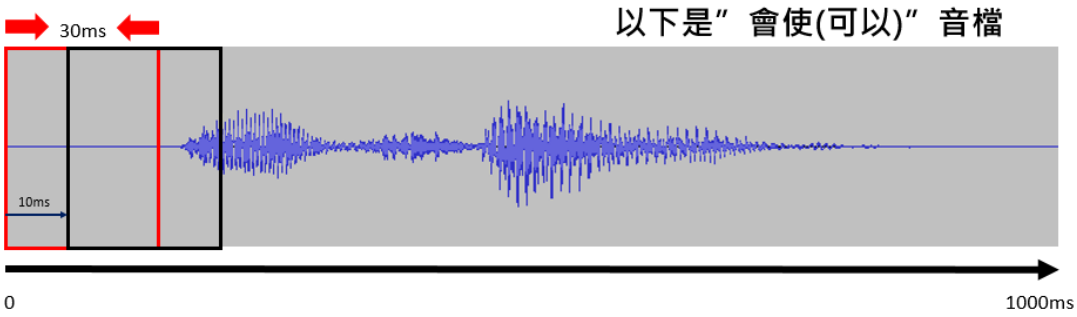
首先，我們為台語辨識研究者建立一個能夠收集音檔的平台，並針對收集到的音檔進行單詞語音辨識的開發，目前我們可以辨識的字共有 30 個單詞如下表所示，而選這三十個詞是因為我們依照 TensorFlow speech command 所使用的 30 單字翻譯成台語，其中包含了十個數字(零到九)及其他日常生活中常用的單詞並提供使用者透過我們建立的平台做單詞音檔即時辨識。

華語	台語	台羅	華語	台語	台羅	華語	台語	台羅
零	零	lîng	上	起去		開	開	khai
一	一	tsi̍t	下	落來	lo̍h-lâi	關	關	kuainn
二	兩	nn̄g	左	倒邊		不可	袂使	b ē -sái/bu ē -sái
三	三	sann	右	正邊		可以	會使	ē -sái
四	四	sì	去	去	khì	志明	志明	
五	五	g ō o	床	眠床	bîn-tshn̂g	春嬌	春嬌	
六	六	la̍k	狗	狗	káu	快樂	快樂	khuài-lo̍k
七	七	tshit	鳥	鳥	tsiáu	房屋	厝	tshù
八	八	peh/pueh	貓	貓	niau	前進	進前	tsìn-tsîng
九	九	káu	樹	樹	tshi ū	後退	退後	thè- ā u

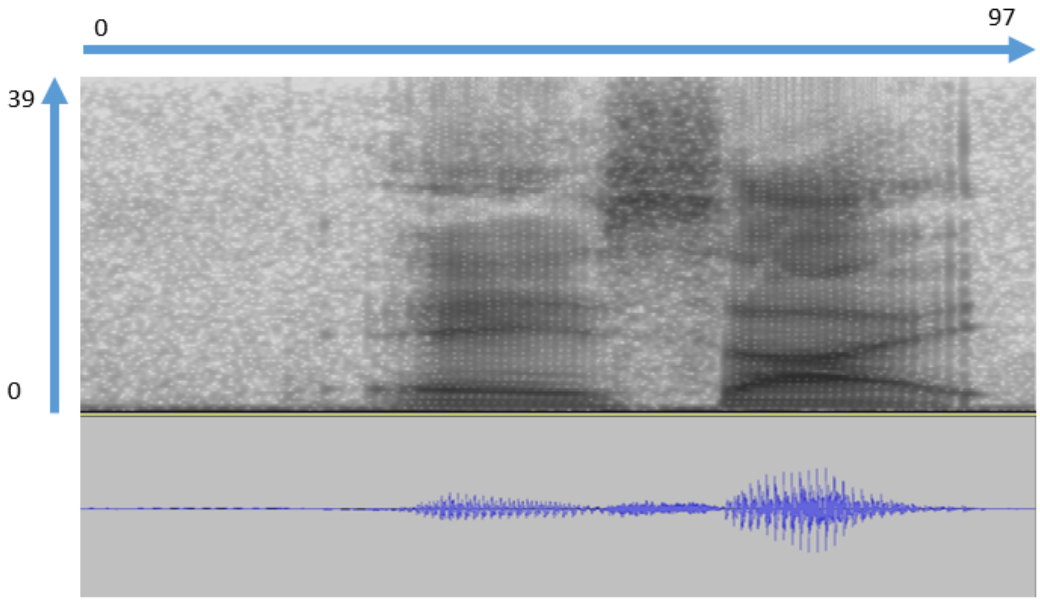
表一、單詞與台語羅馬拼音對照表

(一) 辨識部分

首先在音檔處理部分，我們將這些音檔經過 MFCC 轉換強調人聲的部分後生成頻譜利用 TensorFlow 創建 CNN 模型把頻譜丟入 CNN 進行分類。在音檔格式方面，我們使用 wav 檔 16K 的 sampling rate，即每 1000ms 採 16K 個訊號。設定每次讀取訊號音檔長度為 30ms，而每次訊號讀取移動範圍為 10ms（圖二）。並將讀取到的音檔進行 MFCC 轉換，取 MFCC 特徵點。再來將音檔處理變成頻譜（圖三）。



圖一、音頻圖

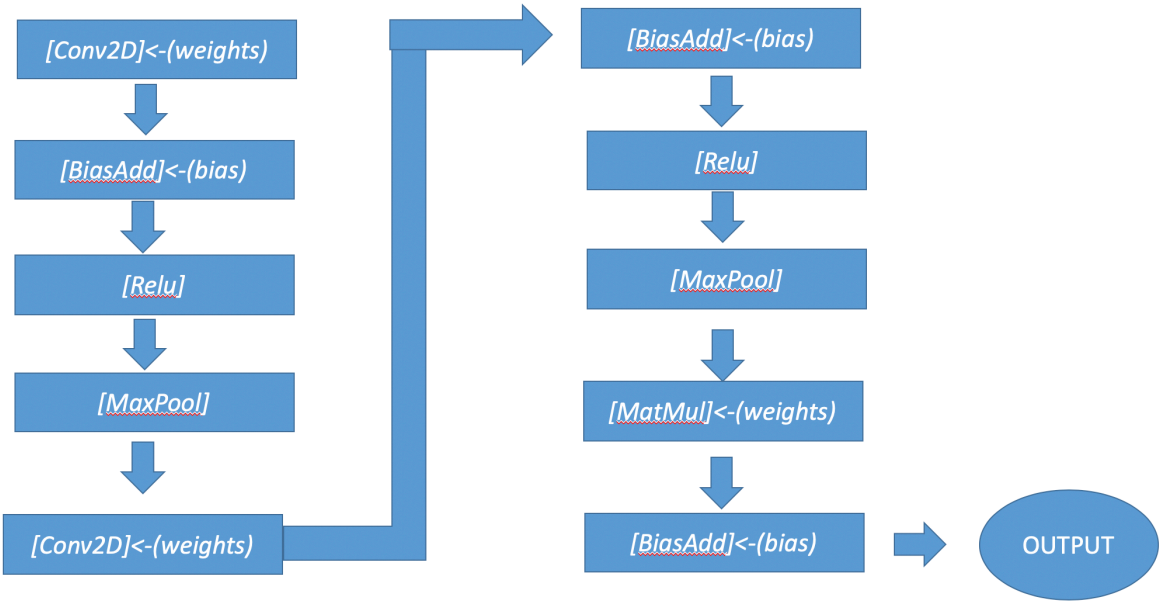


圖二、頻譜圖

(二)CNN 架構

我們選擇使用 CNN 進行訓練，會選擇 CNN 的原因爲通常情況下，語音識別都是基於音頻分析後(MFCC 轉換)的語音頻譜圖完成的，而其中語音頻譜圖是具有結構特點的(各個不同的字音有不同的能量分佈範圍)。要想提高語音識別率，就是需要克服語音信號所面臨各種各樣的多樣性，包括說話人的多樣性(聲音三要素中的音品)，環境的多樣性等(安靜或吵雜的背景音)。而在卷積神經網絡可以達到時域以及頻域的平移不變性(Translation Invariance)，利用此特性將卷積神經網絡應用到語音識別的聲學建模中，則可以此點來克服語音信號本身的多樣性。從這個角度來看，則可以認為是將整個語音信號分析得到的頻譜圖當作一張圖像一樣來處理，採用圖像識別中廣泛應用的

深層卷積神經網絡對其進行識別。而在此我們所採用的架構為 [1]中的 'cnn-trad-fpool3'，因原本架構較為複雜，在此我們將其改為兩層的 Conv2D 跟 MaxPool，詳細結構如下圖所示:



圖三、CNN 結構圖

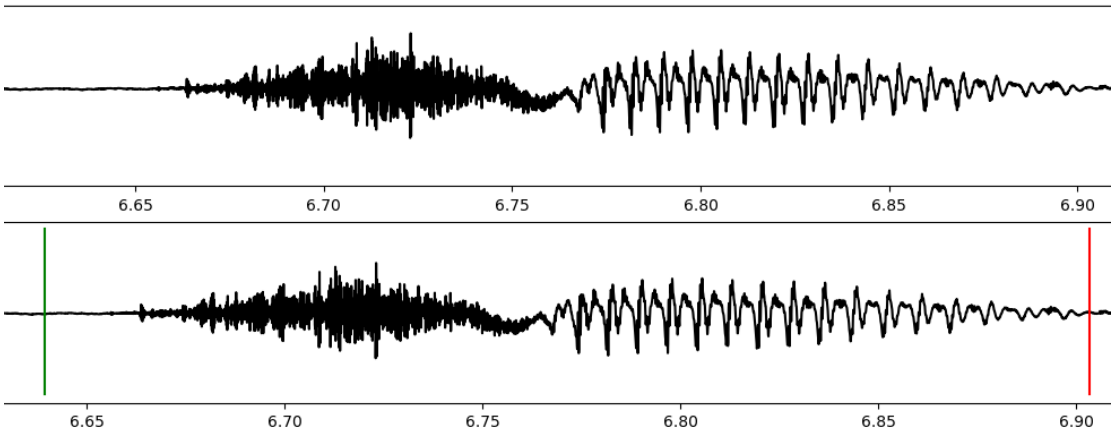
(三)訓練與資料集

市面上並無開源的台語音檔資料，因此我們利用自己架設的網站收集了約 15000 筆台語語音資料，每筆一秒，我們將 15000 筆語音資料分為 3 個子集合 Training data 80%、Validation data 10%、Testing data 10%

(四)端點偵測

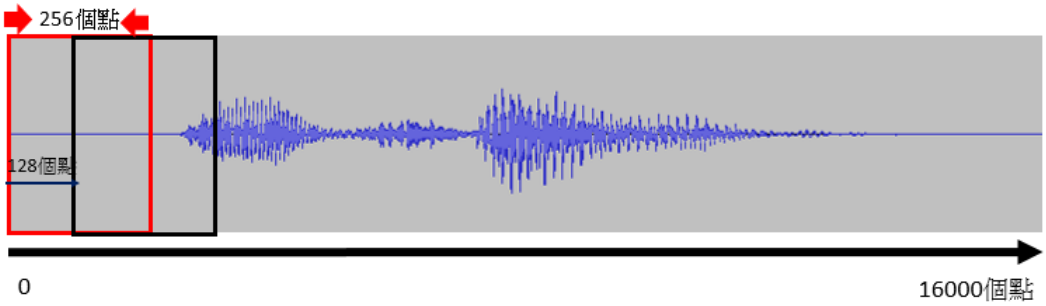
要做到一句話的辨識，會需要做到音檔的分割，從較簡單的連續多個單詞著手，對連續的單詞進行分割，以達成辨識多個單詞。我們使用端點偵測(Endpoint Detection)，定音訊開始和結束的位置，如下圖五所示

我們將端點偵測分為三個部分 1.頻譜正規化:計算分貝數，並給定門檻值，取頻譜中大於門檻值的範圍，認定為有聲音，因此該門檻值能夠決定端點。2.氣音強化: 對於氣音部分，較容易小於門檻值而被摒除在範圍外，所以對頻譜做多次微分，藉此可以將氣音部分凸顯，再經過門檻值的篩選，即可包含較完整的音訊。



圖四 標定選取範圍

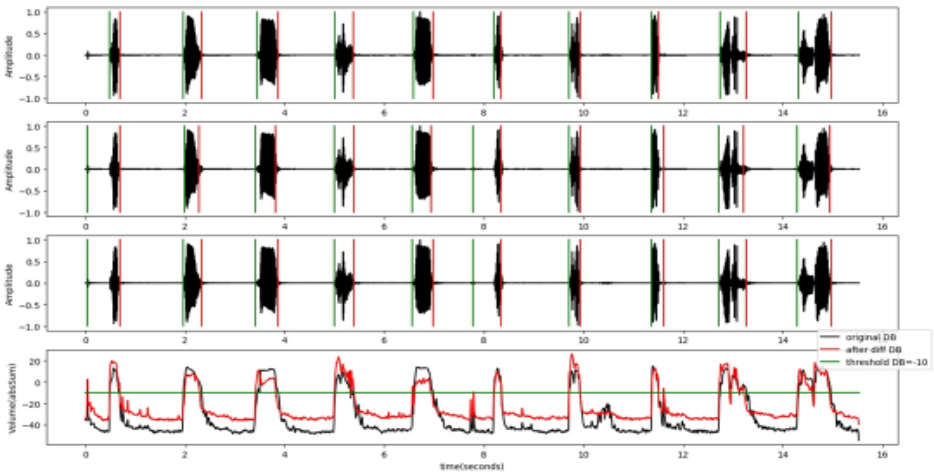
3.後處理: 比較原始頻譜端點及微分後頻譜的端點，並對端點間距做判斷篩選，過濾掉時間較短視為雜音的部分，切割出每個詞彙，並將未滿 1 秒的詞彙補至 1 秒，之後對每個切割出的音檔做辨識。首先我們對音檔進行處理。對頻譜做標準化，將每個點的振幅轉換至[-1,1]每個 frame 取 256 個點，並且每 128 個點作一次平移，計算每個 frame 的 DB 值而我們 DB 值計算方法為對 frame 中每個點做零校正，取平方和後取對數再乘以 10 以 DB=0，DB=-10 等做門檻值進行分割，以下為標準化之示意圖（圖六）



圖五 標準化

圖七是我們切割之結果展示圖，第一列圖為沒有四次微分(將頻譜對時間做四次微分)的圖，而第二列為經過四次微分的圖。第三張是經過後處的圖。第四張是將 DB 門檻加入的展示圖。而綠線為音檔起始線而紅線為音檔結束線。若切割音檔會從綠線到紅

線辨識成一個音檔進行切割，根據實驗結果，經過四次微分後與 DB 值門檻所得結果可以有最大程度的包含進有效頻譜。



圖六 各類切法比較圖

六、 程式環境與訓練結果

我們使用下列規格的電腦配置進行 CNN 模型的訓練

CPU	I7-9800X
MotherBoard	Asus WS X299 Pro
RAM	Kingston 16G*2
SSD	WD Blue 500G
GPU	RTX 2080ti
OS	Ubuntu 16.04
TensorFlow-gpu	1.13.0rc2

表二、環境配置

利用 confusion matrix 顯示訓練結果（圖八），在 CNN 架構中，目前所能辨識的正確率為 90%

```
INFO:tensorflow:Confusion Matrix:
 [[254   0   0 ...   0   0   0]
 [  0   0   0 ...   0   0   0]
 [  0   0  80 ...   0   0   0]
 ...
 [  0   0   0 ...  82   0   0]
 [  0   0   0 ...   0  58   0]
 [  0   0   0 ...   0   0  50]]
INFO:tensorflow:Final test accuracy = 90.1% (N=2789)
```

圖七、訓練結果

在實際辨識上，我們有撰寫出一個網頁可以讓使用者選擇連續辨識(可辨識詞的範圍在30 詞內，圖九)，以及一次一個單詞辨識(錄音一秒，圖十)

圖八、連續辨識

圖九、單詞辨識

從圖九與圖十中右半部皆有上傳、下載與刪除之按鈕，這裡就是緒論中有提到的，我們希望可以製作一個平台，讓使用者可以在使用我們辨識程式時，可以上傳辨識中所錄下的音檔，也可以將錄音下載回自己電腦，如不克上傳，也可以按下刪除鍵。我們希望成為一個收集平台，保存台語文化。

七、　參考資料

[1] Tara N. Sainath, Carolina Parada, “Convolutional Neural Networks for Small-footprint Keyword Spotting,” *INTERSPEECH 2015*, 2015.

[2] P. Warden, “Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition,” *Google Brain Mountain View*, California, April,2018.