國立台北大學資訊工程學系專題報告

中文語音大師：透過深度學習自動校正發音的學習系統

Chinese Pronunciation Master

專題組員: 呂冠廷，曹禕中，呂詠儒

專題編號:PRJ-NTPUCSIE-113-004

執行期間:113 年 7 月 至 114 年 6 月

1. 摘要

隨著全球化的發展,國際交流日益頻繁。在這個時代，不僅國人需要學習英語，許多外國人同樣開始學習中文。而在學習中文的過程中，口語練習是一個至關重要的環節。許多外國人在說中文時常出現奇怪的口音,而對於初學者來說，若缺乏教師的指導，往往難以察覺自己的發音錯誤，即便察覺了，也未必能明白錯誤所在或知道如何修正。現實中,大部分學習者都無法隨時獲得真人指導，且**市面上並無類似功能之軟體，即便是最類似的 Google 翻譯對於不熟悉中文的使用者辨識效果也不佳**。因此，我們計劃開發一套基於深度學習的中文發音矯正系統,**能辨識使用者所說的任意語句，而非僅限特定句子**。系統將判斷使用者的語意，並協助其糾正發音，促進更有效的中文學習。

本計畫預計開發的系統主要流程如下：首先，我們訓練了兩套自動語音辨識(ASR)模型，將語音輸入以字元和單字這兩種形式轉換為對應的拼音序列。接著，透過 n-gram 和 beam search 等演算法，初步修正辨識出的語音結果。隨後,我們使用 OpenAI 的 API，將拼音轉回中文。同時，在聲調辨識方面，我們利用 CTC Greedy 解碼將音訊切割為單字單位，再擷取每個音節的梅爾頻譜圖 (Mel-spectrogram) 作為特徵，並利用 CNN 模型判斷其對應的聲調，以分析使用者所發出的每個音節的聲調。最後，透過對比使用者的發音與正確句子的發音（聲調與念法），找出發音錯誤。系統將以直觀的方式顯示使用者的發音及聲調錯誤所在。**本系統不僅能處理任意語句，亦能結合語意理解與聲調分析提供多層回饋，在現有中文學習工具中展現出高度創新性。**我們相信，以拼音（英文字母）的形式呈現發音資訊，更能幫助外國學習者理解和掌握中文發音。

2. 簡介

近年來，市面上雖然出現了一些可以協助學習英文的 AI 軟體，但我們發現，針對外國人學習中文、特別是中文發音與聲調矯正的工具仍十分稀少，僅有的多數仍為真人線上教學，或僅支援固定語句。現有的語音學習工具，例如 **Google 翻譯或 DeepL 翻譯，不僅無法辨識初學者的不標準中文，而且也無法提供針對發音錯誤與聲調偏誤的回饋機制**。這成為現今語音學習者想要自由學習中文過程中的重大限制。因此，我們想要設計一款基於深度學習的中文發音矯正系統，

能辨識使用者所說的任意語句，並針對發音與聲調錯誤提供即時回饋。

本專題旨在研發一套針對中文初學者所設計的發音矯正系統，結合了自動語音辨識(ASR)、語言模型(LLM)與聲調分析等深度學習技術。與現有僅具陪聊或翻譯功能的中文學習工具相比，本系統最大的特色在於能辨識使用者所說的任意語句，並從拼音準確性與聲調辨識兩個層面，判斷使用者的實際發音與標準發音間的差異。

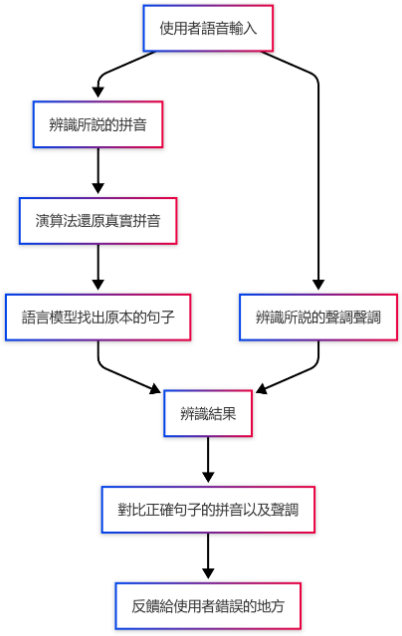


圖 1: 系統流程總覽圖

本專題完成後，預期將在中文發音學習領域產生以下效益：

[1] 學習者可不依賴真人教師，即時獲得拼音與聲調的矯正建議。

[2] 系統**可辨識任意語句**，提供精準回饋，不受限於固定語句，提升彈性與實用性。

[3] 推動中文教學，讓學習中文口音門檻降低。

[4] 系統具良好擴展性，未來可應用於非標準發音辨識等更多領域

3. 專題進行方式

3.1 人員分工：

我們分工方式是定期討論整體系統架構，並各自設計能達成階段性目標的部件，最後再整合成一個軟體。

1. 呂冠廷：演算法設計、模型研發探索，最終架構之所有模型研發與訓練，系統優化，資料收集測試。
2. 曹禕中：模型研發探索、前端、後端模型部署與系統優化。
3. 呂詠儒：前端設計、翻譯功能，多語言界面、語音與錄音功能，資料測試。

3.2 演算法設計

本系統實作之演算法主要分爲四個部分, 整體流程圖如圖 2 所示。

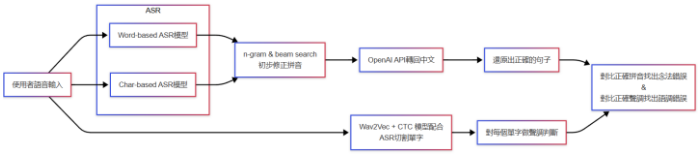


圖 2:演算法整體流程圖

首先，系統會接收使用者輸入的語音訊號，並分別送入「字元層級 ASR 模型 (Char-based ASR)」及「單字層級 ASR 模型 (Word-based ASR)」進行拼音辨識，同時利用梅爾頻譜圖 (Mel Spectrogram) 提取語音中的聲調資訊。接著，系統將透過 n-gram 和 beam search 演算法對拼音結果進行初步修正，輸出多個可能的拼音序列並進行評分排序。修正後的拼音序列會被輸入至 OpenAI API，利用提示工程還原出使用者想發出的中文語句。最後，系統將對照標準發音拼音及聲調，標示出學習者的發音與語調錯誤，並以直觀且易理解的方式告訴使

用者念法以及聲調上存在的錯誤。以下針對計畫預計開發之系統演算法進行細節介紹。

第一部分：口語語音識別出拼音字串

我們訓練了兩個 model,「字元層級 ASR 模型 (Char-based ASR)」及「詞彙層級 ASR 模型 (Word-based ASR)」，來進行拼音辨識。其中兩個 model 皆使用 Connectionist Temporal Classification Loss 進行訓練，以對齊長度不匹配的音訊序列與目標文字序列。CTC Loss 透過以下公式計算：

$$\mathcal{L}_{CTC} = -\log\left(\sum_{a\in\mathcal{A}(y)}\prod_{t=1}^{T} P(a_t|x)\right)$$

該方法之架構圖:

圖 3: CTC 拼音輸出流程圖

字母和單字層級的識別差異如下：

a. 字母層級：利用 AIShell 中文語音，NexData 模糊中文，FAD 等資料集來達成把中文音頻以字母對字母的方式轉換成拼音。CTC 預測的輸出空間為 27，也就是 26 個英文字母和 1 個空格。

b. 單字層級：模型為開源模型 Wav2Vec2-Large-XLSR-53 對單字拼音做微調訓練，CTC 預測的輸出空間為自建的拼音對應表(word map)，共包含 404 種拼音類別。Word map 計算出現頻率的公式：

$$f(w) = \sum_{i=1}^{M} \delta(w_i, w)$$

其中，w_1 到 w_M 代表輸入之資料集所有單字的拼音。

第二部分：拼音字串錯誤分析與修正

1. 使用 Levenshtein Distance 修正字詞

ASR 模型輸出後，我們先以萊文斯坦距離修正無效拼音，將其替換為 word map 中最相近的合法拼音。該距離透過計算「插入（Insertion）」、「刪除（Deletion）」與「替換（Substitution）」所需的最少步驟數定義，如下：

$$D(i,j) = \begin{cases} \max(i,j) & \text{若 } \min(i,j) = 0 \\ \min\begin{cases} D(i-1,j)+1 \\ D(i,j-1)+1 \\ D(i-1,j-1)+1_{[w_i \neq v_j]} \end{cases} \end{cases}$$

2. Beam Search 解碼

隨後，我們利用 Levenshtein Distance(由近到遠)和詞頻 (透過百萬字的中文資料集所建立的 word map 決定)去延伸出 7 個可能的候選單字字串。由此字串與其後單字字串每次共可延伸出最多 73 個可能的候選單字陣列，再利用 n-gram 語言模型(設 n=3)，為每個候選單字陣列評分，最後保留前 5 條分數最高之候選序列。該部分演算法過程如圖 4 所示。

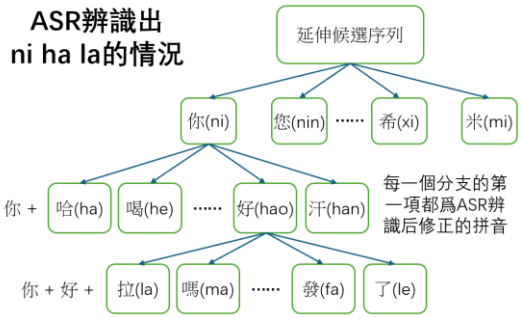


圖 4: 第二部分演算法示意圖

其中 Beam Search 的評分方式為:

$$S(y) = \sum_{t=1}^{T} \log P_{ASR}(y_t|x) + \lambda \cdot \log P_{LM}(y)$$

和用來評估候選序列的語言合理性之三元語言模型 (Trigram)定義如下：

$$P(w_1, w_2, ..., w_T) = \prod_{t=1}^{T} P(w_t|w_{t-2}, w_{t-1})$$

每個條件機率由以下公式計算：

$$P(w_t|w_{t-2}, w_{t-1}) = \frac{C(w_{t-2}, w_{t-1}, w_t) + 1}{C(w_{t-2}, w_{t-1}) + |V|}$$

第三部分：還原中文字串

利用前文的方法，我們已經利用了語意分析出了最有可能的五個發音序列。在此步驟，我們會利用提示工程，把已經還原到正確或者非常接近正確的拼音輸出給 OpenAI 的 API, 讓其返還正確的中文語句。該方案經過微調提示詞。

第四部分：單字聲調辨識

我們先通過單獨訓練好的 Wav2Vec2 + CTC 模型，取得每個時間點的 token 機率分佈，並以 CTC Greedy 解碼法得到每個單字的獨立音頻。

其中，中文的聲調主要可以分爲五種聲調，各自有明顯頻率走向差異。我們從梅爾頻譜圖提取特徵來產生特徵向量 x：

$$\mathbf{x} = [\bar{F0}, F0_{max}, F0_{min}, \text{slope}, \text{variance}]$$

最後，我們利用 TonePerfect 資料集訓練了一個 CNN 音訊分類模型，辨識每個單字的聲調類型。輸入為單字音頻，並轉換為梅爾頻譜圖（Mel-spectrogram）作為模型的特徵向量。模型輸出為四類分類結果，分別對應中文的一聲至四聲。其中，在標準發音中，各聲調在頻率走向上展現出顯著差異，其變化特性如下圖所示：

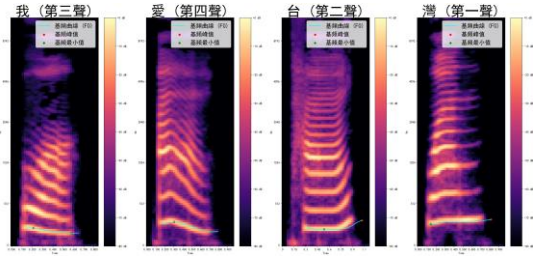


圖 5: 聲調走向示意圖

研發本系統遇到的困難

本專題主要遇到了演算法設計和系統優化的問題。因爲不清楚如何在不知道正確答案的前提下辨識出發音錯誤，所以我們在設計演算法上花了很多時間探索，也設計出了許多最後沒有用到的模型。而在初步設計完系統後，系統的運行速度也相對較慢，我們花了不少時間才將其優化到可以快速辨識且標注錯誤的發音。

4. 主要成果與評估

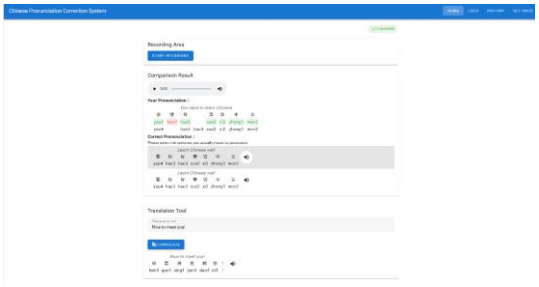


圖 6: 系統界面

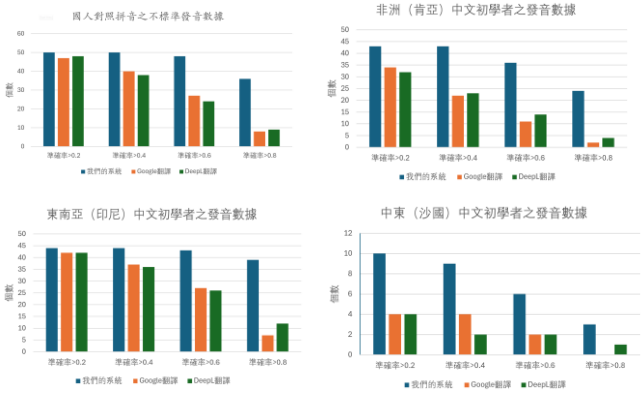
上圖便是我們目前的系統界面，包含了多項輔助學習功能，例如翻譯查詢、歷史記錄、標準發音的 TTS 播放，多種語言界面來輔助學習者進步。

其中，我們收集了來自世界各地中文初學者的語音資料來測試我們的系統語音辨識度。方式為提供拼音文字，並請受試者依據拼音朗讀多組中文句子。在得到音頻后，我們會測試我們系統的辨識結果，並於 Google 翻譯以及 DeepL 翻譯進行對比。分數計算方法為對比識別語句和正確語句之 edit distance。測試結果如下：

發音辨識準確率總體平均分數			
發音地區	我們的系統	Google翻譯	DeepL翻譯
國人發音	0.90	0.57	0.56
非洲發音	0.79	0.40	0.42
東南亞發音	0.91	0.61	0.61
中東發音	0.67	0.27	0.22

該分數是以小數點表示百分比，舉例而言，0.9 代表著結果與正確句子總體只差了 10%的單字距離。在整體平均準確率方面，我們的系統超越市

面主流中文辨識系統(Google 翻譯與 DeepL) 30% 以上，展現出本系統在處理非標準中文發音時的穩定性與辨識精度。我們也分別對每個地區的辨識語句做了纍計準確率比對，如下：



如圖標所示，對於不論是東南亞，非洲，或者是在中東地區的中文初學者，我們的系統皆遠優於 Google 與 DeepL，展現出更強的跨地區發音辨識穩定性與準確性。值得注意的是，本系統在準確率高於 80%的高標準區間中表現明顯領先。

我們有將所收集之測試資料集上傳，我們的系統也支持隨時重新測試以查驗該組數據的準確性。數據已經保留與 Google 雲端上，鏈接為：https://drive.google.com/drive/folders/1DijrcKCNinRYJ76l4Aq9Zkq06kSXVlfA?usp=sharing

我們的系統可以於手機端和電腦端的網頁使用。

5. 結語與展望

本專題成功設計並實作了一套結合 ASR、聲調分類與語言模型的中文發音矯正系統。透過聲音切割與語調分析技術，**系統能有效辨識使用者任意語句的發音偏誤，並提供拼音與聲調兩層級的即時回饋**。根據測試結果，我們的系統對於不同地區的中文初學者的語音數據，在**平均準確率方面都比 Google翻譯和 DeepL 翻譯高出至少 30%，且在「準確率 > 0.8」的高標準區間中亦明顯領先**。這顯示本系統在拼音辨識與發音糾錯上，相較於市面翻譯工具更具發音識別準確度。更重要的是，不同於翻譯軟件，本系統可準確標示出使用者的口音錯誤位置，且有多項輔助學習功能。

未來，我們將考慮導入更多輔助學習功能，比如：**利用目前已實現的單字級音訊切割能力，讓系統在偵測到使用者發音錯誤時，直接播放該單字的「使用者實際發音」與「標準發音」作為對照**，讓學習者能夠單獨聽出該單字的發音差別。此外，我們也將持續優化我們模型，進一步提升整體辨識準確率與即時反應能力，打造更智慧化且實用的中文發音學習工具。

6. 銘謝

感謝來自肯亞的 Jerry, 來自印尼的 Jason, 來自中東的 Sarah 所提供寶貴的成果評估資料集。他們都在不熟悉甚至完全不會中文的前提下，看著拼音說了許多語句。感謝所有臺北大學資訊工程學系的教授，給予了我們可以完成這個專題所需要的知識。

7. 參考文獻

[1] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, "Unsupervised Cross-lingual Representation Learning at Scale", arXiv preprint arXiv:2006.13979, 2020. [Online]. Available: https://arxiv.org/abs/2006.13979

[2] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist Temporal Classification: Labelling

Unsegmented Sequence Data with Recurrent Neural Networks," in Proceedings of the 23rd International Conference on Machine Learning (ICML), Pittsburgh, PA, USA, 2006. [Online]. Available: https://www.cs.toronto.edu/~graves/icml_2006.pdf

[3] Y.-D. Shieh, "wav2vec2-large-xlsr-53-chinese-zh-cn-gp," Hugging Face, 2021. [Online]. Available: https://huggingface.co/ydshieh/wav2vec2-large-xlsr-53-chinese-zh-cn-gp

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv:1810.04805*, 2018. [Online]. Available: https://arxiv.org/pdf/1810.04805