Computational Linguistics and Chinese Language Processing
Vol. 26, No. 1, June 2021, pp. 17-32

NSYSU-MITLab 團隊於福爾摩沙語音辨識競賽 2020 之語音辨識系統

NSYSU-MITLab Speech Recognition System for Formosa Speech Recognition Challenge 2020

林洪邦*、陳嘉平*

Hung-Pang Lin and Chia-Ping Chen

摘要

本論文中，我們描述了 NSYSU-MITLab 團隊在福爾摩沙語音辨識競賽 2020 (Formosa Speech Recognition Challenge 2020, FSR-2020) 中所實作的系統。我們使用多頭注意力機制 (Multi-head Attention) 所構成的 Transformer 架構建立了端到端的語音辨識系統，並且結合了連續性時序分類 (Connectionist Temporal Classification, CTC) 共同進行端到端的訓練以及解碼。我們也嘗試將編碼器更改為結合卷積神經網路 (Convolutional neural network, CNN) 與多頭注意力機制的 Conformer 架構。同時我們也建立了深度神經網路結合隱藏式馬可夫模型 (Deep Neural Network-Hidden Markov Model, DNN-HMM) ，其中我們以時間限制自注意力機制 (Time-Restricted Self-Attention, TRSA) 及分解時延神經網路 (Factorized Time Delay Neural Network, TDNN-F) 建立深度神經網路的部分。最終我們在台文漢字任務上得到最佳的字元錯誤率 (Character Error Rate, CER) 為 43.4% 以及在台羅拼音任務上取得最佳的音節錯誤率 (Syllable Error Rate, SER) 25.4%。

Abstract

In this paper, we describe the system team NSYSU-MITLab implemented for Formosa Speech Recognition Challenge 2020. We use the Transformer architecture

* 國立中山大學資訊工程學系
Department of Computer Science and Information Engineering National Sun Yat-sen University
E-mail: m083040013@g-mail.nsysu.edu.tw; cpchen@mail.cse.nsysu.edu.tw

composed of Multi-head Attention to construct an end-to-end speech recognition system and combine it with Connectionist Temporal Classification (CTC) for end-to-end training and decoding. We have also built a deep neural network combined with a hidden Markov model (DNN-HMM). We use Time-Restricted Self-Attention and Factorized Time Delay Neural Network (TDNN-F) for the deep neural network in DNN-HMM. The best performance we have achieved with the proposed methods is the character error rate of 45.5% for Taiwan Southern Min Recommended Characters (台文漢字) task and syllable error rate 25.4% for Taiwan Minnanyu Luomazi Pinyin (台羅拼音) task.

關鍵詞：自動語音辨識、Transformer、Conformer、連續性時序分類、聲學模型

Keywords: Automatic Speech Recognition、Transformer、Conformer、Connectionist Temporal Classification、Acoustic Model

1. 緒論 (Introduction)

近年來由於硬體效能的提昇以及深度學習 (Deep Learning) 理論的蓬勃發展，人工智慧成為了熱門的研究議題，無論在圖像、文字、語音等應用上皆可看到其蹤影。自動語音辨識 (Automatic Speech Recognition, ASR) 是深度學習中一個重要的應用， 在過往經常使用隱藏式馬可夫模型-高斯混合模型 (Hidden Markov Model-Gaussian Mixture Model, HMM-GMM) 來建立語音辨識模型，之後隨著深度學習的發展，出現了結合深度神經網路 (Deep Neural Network) 的 DNN-HMM。時延神經網路 (Time Delay Neural Network, TDNN) 以及長短期記憶 (Long Short-Term Memory, LSTM) 所構成的神經網路應用在 DNN-HMM 上獲得了不錯的成效(Peddinti *et al.,* 2018)，而時間限制自注意力機制 (Time-Restricted Self-Attention, TRSA) 也被證實可以用來代替網路中的 TDNN 或 LSTM (Povey *et al.,* 2018)。近年來自動語音辨識中端到端 (End-to-End) 方法成為了另一個熱門的研究項目，端到端的語音辨識系統能夠將聲音訊號輸入至一個模型便可直接輸出對應的文字序列。連續時序性分類 (Connectionist Temporal Classification, CTC) (Graves *et al.,* 2006)以及基於多頭注意力機制 (Multi-head Attention) 的 Transformer 架構 (Vaswani *et al.,* 2017) (Dong *et al.,* 2018)皆可做為端到端語音辨識的模型，其中又出現了結合兩者的目標函數共同進行訓練混合模型(Karita *et al.,* 2019)。

現今的語音辨識系統在英文以及中文等資源豐富的語言上已經能達到很好的辨識效果了，但是對於台語等資源相對較稀少的語言卻尚未有足夠的研究。我們參加了台語語音辨識競賽 FSR-2020 (Liao *et al.,* 2020)，並針對台文漢字任務建立了端到端的台語語音辨識模型，以及針對台羅拼音數字調任務建立了 DNN-HMM 以及端到端的台語語音辨識模型。本文分為四個部分：第一部分為緒論；第二部分為研究方法，介紹端到端的語音辨識模型架構以及 DNN-HMM 的模型架構；第三部分為實驗，介紹資料集、實驗設置以及實驗結果；第四部分為結論。

2. 研究方法 (Research Methods)

2.1 端到端模型架構 (End-to-End Model Architecture)

2.1.1 Transformer編碼器 (Transformer Encoder)

我們使用的端到端模型架構由 Transformer 的編碼器以及解碼器所構成，並且利用編碼器的輸出計算連續時序性分類 (Connectionist Temporal Classification, CTC)，模型架構如圖 1 所示。我們使用的聲學特徵是 80 維的 Fbank (Filter bank) 加上 3 維的音調 (Pitch)，首先我們會先以兩層 att 個卷積核 (kernel)、步長 (stride size) 2 、卷積核大小為 3 的卷積神經網路 (Convolutional neural network, CNN) 降低輸入特徵的維度，得到新的特徵 $X^{sub} \in R^{seq \times att}$ ，其中 att 為注意力機制的特徵維度大小。第 i 層編碼器計算流程如下：

$$X_0 = X^{sub} + PE \tag{1}$$

$$X_i' = \text{Layernorm}\big(X_i + \text{MHA}(X_i, X_i, X_i)\big) \tag{2}$$

$$X_{i+1} = \text{Layernorm}\big(X_i' + \text{FF}(X_i')\big) \tag{3}$$

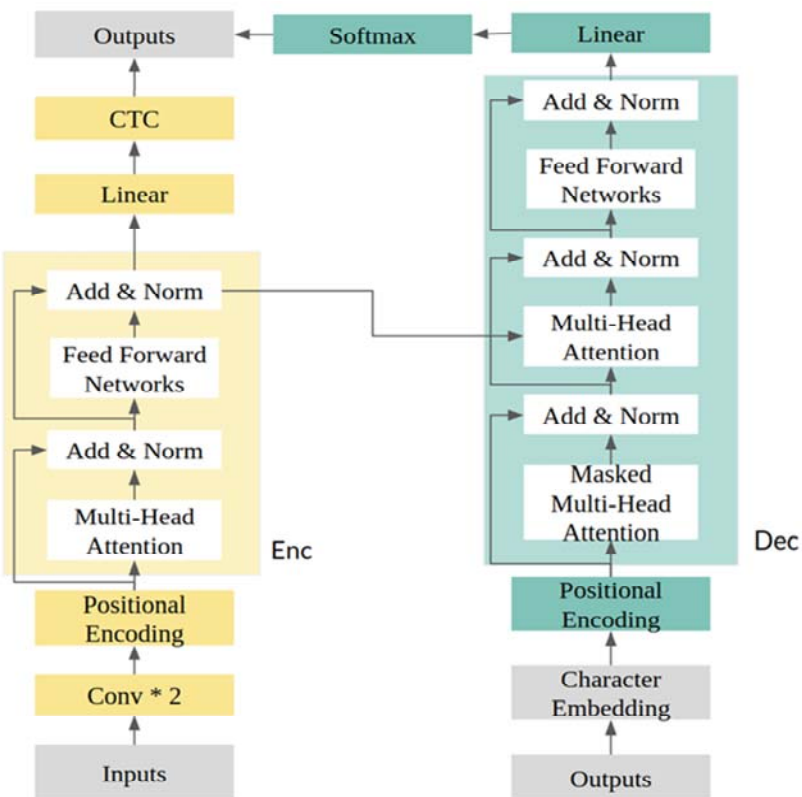

圖 1. Transformer-based 模型架構
[Figure 1. Transformer-based model architecture]

由於多頭注意力機制無法取得輸入序列的前後位置資訊，因此輸入編碼器的特徵要再額外加上位置編碼 PE (Positional Encoding) (Vaswani *et al.*, 2017)。多頭注意力機制 MHA 的計算方式如下：

$$\text{attention}(Q,K,V) = \text{softmax}\left(\frac{QK^{\top}}{\sqrt{att}}\right)V \tag{4}$$

$$H_h = \text{attention}\left(QW_h^q, KW_h^k, VW_h^v\right) \tag{5}$$

$$\text{MHA}(Q,K,V) = [H_1, H_2, \dots, H_{head}]W^{head} \tag{6}$$

其中 $K,V \in R^{k\times att}$ 以及 $Q \in R^{q\times att}$ 是 MHA 層的輸入，$W_h^q, W_h^k, W_h^v \in R^{att\times(att/head)}$ 以及 $W^{head} \in R^{att\times att}$ 是可以學習的參數，att 為注意力機制的特徵維度大小。head 個自注意力機制計算的結果 H_h 將串接起來，並與 MHA 的輸入進行殘差連結 (Residual connect) 後再通過層標準化 (Layer Normalization) (Ba *et al.*, 2016)得到最終的輸出。公式 (3) 中的 FF 則是前饋神經網路 (Feed-Forward Neural Network)，由兩層全連接層以及 Rectified Linear Unit (ReLU) 激活函數所構成。在經過數層編碼器的運算後，得到最終的編碼器輸出 $X_e \in R^{seq\times att}$。

2.1.2 Transformer 解碼器 (Transformer Decoder)

得到 Transformer 編碼器的輸出 X_e 後，Transformer 解碼器會以 X_e 以及前一個時刻的字元序列 $Y[1:u]$ 計算下一個輸出字元 $Y[u+1]$ 的機率。第 j 層解碼器的計算流程如下：

$$E = \text{Embed}(Y[1:u]) \tag{7}$$

$$Z_0 = E + PE \tag{8}$$

$$Z_j' = \text{Layernorm}\left(Z_j + \text{MHA}(Z_j, Z_j, Z_j)\right) \tag{9}$$

$$Z_j'' = \text{Layernorm}\left(Z_j' + \text{MHA}(Z_j', X_e, X_e)\right) \tag{10}$$

$$Z_{j+1} = \text{Layernorm}\left(Z_j'' + \text{FF}(Z_j'')\right) \tag{11}$$

其中公式(7) 會將字元序列轉換為詞嵌入 (Word Embedding) $E \in R^{u\times att}$，加上位置編碼後做為第一層解碼器的輸入 Z_0。下一個輸出字元 $Y[u+1]$ 的後驗機率計算方式如下：

$$[p_{s2s}(Y[2]|Y[1], X_e), \dots, p_{s2s}(Y[u+1]|Y[1:u], X_e)] = \text{softmax}(Z_d W^{att} + b^{att}) \tag{12}$$

$$p_{s2s}(Y|X_e) = \prod_u p_{s2s}(Y[u+1]|Y[1:u], X_e) \tag{13}$$

其中 Z_d 為最後一層解碼器的輸出，$W^{att} \in R^{att\times token}, b^{att} \in R^{token}$ 則是可以學習的參數，在經過 softmax 的運算後可以得到每個可能字元的輸出機率。

2.1.3 Conformer編碼器 (Conformer Encoder)

除了 Transformer 編碼器外，我們也嘗試使用結合 CNN 與 Transformer 的 Conformer 編碼

器 (Gulati *et al.,* 2020)，透過卷積模組 (Convolution Module) 以及 多頭注意力機制，Conformer 能更好的取得局部以及全局的訊息，模型架構如圖 2 所示。第 i 層 Conformer 編碼器的計算流程如下：

$$X_i' = X_i + \frac{1}{2}\mathrm{FF}(\mathrm{Layernorm}(X_i)) \tag{14}$$

$$X_i'' = X_i' + \mathrm{MHA}(\mathrm{Layernorm}(X_i')) \tag{15}$$

$$X_i''' = X_i'' + \mathrm{Conv}(X_i'') \tag{16}$$

$$X_{i+1} = \mathrm{Layernorm}\left(X_i''' + \frac{1}{2}\mathrm{FF}(\mathrm{Layernorm}(X_i'''))\right) \tag{17}$$

Conformer 使用馬卡龍網路 (Macaron-Net) (Lu *et al.,* 2019)的架構，將兩個前饋神經網路置於編碼器的頭尾，並且將前饋神經網路中的 ReLU 激活函數替換為 Swish 激活函數 (Ramachandran *et al.,* 2017)。另外 Conformer 採用源於 Transformer-XL 的相對位置編碼 (Relative Positional Encoding) (Dai et al. 2019)，相較於一般的位置編碼，相對位置編碼在面對不同長度的輸入能取得較好的位置資訊。在多頭注意力機制之後，Conformer 加入了卷積模組，卷積模組從單位卷積層 (Pointwise Convolution) 以及門控線性單元 (Gated Linear Unit, GLU) (Dauphin *et al.,* 2017)開始，接著會再經過深度卷積層 (Depthwise Convolution)、批量標準化 (Batch Normalization)、Swish 激活函數及單位卷積層。

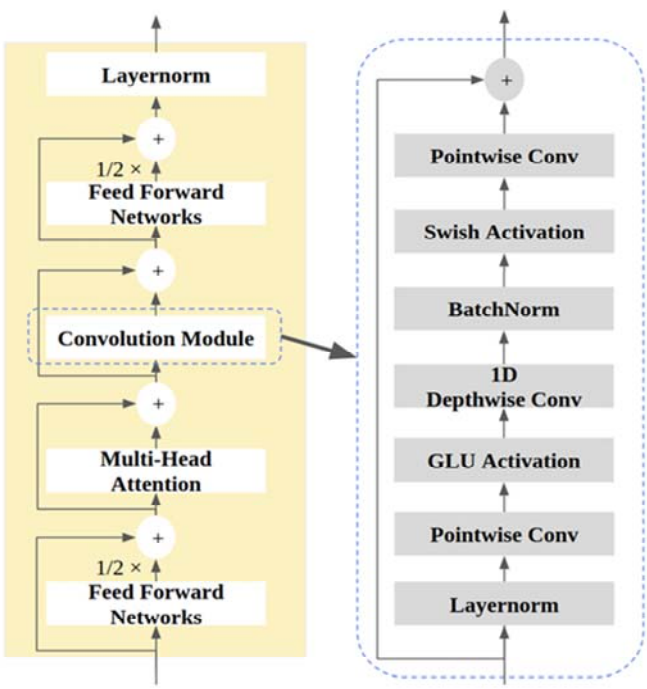

圖 2. Conformer 模型架構
[Figure 2. Conformer model architecture]

2.1.4 混合訓練及解碼 (Hybrid Training and Decoding)

除了解碼器的輸出外，我們也會利用編碼器的輸出計算連續時序性分類 (CTC)，其計算方式如下：

$$C = \text{softmax}(X_e W^{ctc} + b^{ctc}) \tag{18}$$

$$p(\pi|X_e) = \prod_{t=1}^{seq} C[t, \pi[t]] \tag{19}$$

$$p_{ctc}(Y|X_e) = \sum_{\pi \in \beta^{-1}(Y)} p(\pi|X_e) \tag{20}$$

其中 $W^{ctc} \in R^{att \times token}, b^{ctc} \in R^{token}$ 是可以學習的參數。由於輸入序列的長度一般會大於輸出序列的長度，因此輸出會有多種可能的組合，這些組合記為π，公式(19)中 $C[t, \pi[t]]$ 代表 X_e 的第 t 個幀輸出 π[t] 的機率。$\beta(\pi)$ 是一個多對一的函式，會將 π 中冗餘的字元刪除，例如 β(a∅aabb) = aab 。公式(20)中 $\beta^{-1}(Y) = \{\pi | Y = \beta(\pi)\}$ 代表所有可以形成文字序列 Y 的組合。最終我們將混合解碼器以及 CTC 的輸出，得到損失函數：

$$L_{mtl} = -\alpha \log p_{s2s}(Y|X_e) - (1-\alpha) \log p_{ctc}(Y|X_e) \tag{21}$$

其中 α 為超參數，控制解碼器及 CTC 在損失函數中所佔的比例。在解碼階段時，除了解碼器以及 CTC 的輸出外，我們額外加入語言模型共同進行解碼，解碼方式如下：

$$\hat{Y} = \arg\max_{Y \in y^*} \{\lambda \log p_{s2s}(Y|X_e) + (1-\lambda) \log p_{ctc}(Y|X_e) + \gamma \log p_{lm}(Y)\} \tag{22}$$

其中y^* 代表輸出的候選字集合，$p_{lm}(Y)$ 則代表語言模型派給候選序列 Y 的機率。λ 及 γ 為超參數，分別控制在解碼階段時解碼器、CTC 以及語言模型所佔的比例。

2.2 DNN-HMM

2.2.1 分解時延神經網路 (Factorized Time Delay Neural Network)

除了端到端的模型架構外，我們也嘗試針對台羅拼音任務建立 DNN-HMM 的語音辨識模型。在聲學模型中，我們使用到了分解時延神經網路 (Factorized Time Delay Neural Network, TDNN-F) (Povey *et al.*, 2018)，此種模型架構拆解一般時延神經網路的權重矩陣，並在訓練過程中限制其中一個分解的權重矩陣保持正交，以維持訓練時的穩定度。假設 M 為參數矩陣，我們定義 $P \equiv MM^{\mathrm{T}}, Q \equiv P - I$，並希望透過參數更新最小化 $f = \text{tr}(QQ^{\mathrm{T}})$ 使得 M 保持正交。參數更新的公式如下：

$$M \leftarrow M - \frac{1}{2\alpha^2}(MM^{\mathrm{T}} - \alpha^2 I)M \tag{23}$$

其中α 是縮放參數，設為$\sqrt{\text{tr}(PP^{\mathrm{T}})/\text{tr}(P)}$ 。圖 3 為 TDNN-F 的模型架構，1536 維的隱藏層會被拆解為 1536×160×1536 ，其中第一個權重矩陣會保持正交限制，之後再加上 ReLU 激活函數、批量標準化以及殘差連結 。

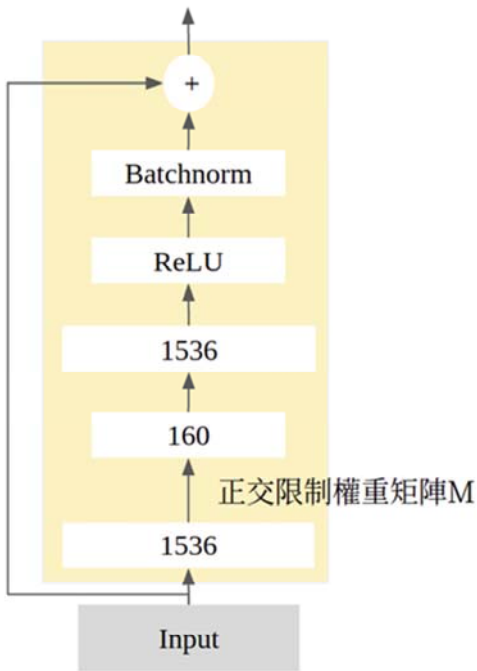

圖 3. TDNN-F 模型架構
[Figure 3. TDNN-F model architecture]

2.2.2 時間限制自注意力機制 (Time-Restricted Self-Attention)

除了 TDNN-F 外，我們也使用了時間限制自注意力機制 (Time-Restricted Self-Attention, TRSA) (Povey *et al.*, 2018)，此種特殊的自注意力機制在計算公式(4)時會限制模型的關注範圍，其計算方式如下：

$$\boldsymbol{y_t} = \sum_{\tau=t-L}^{t+R} c_t(\tau)\boldsymbol{v_t} \tag{24}$$

$$c_t(\tau) = \frac{\exp(\boldsymbol{q_t}\cdot\boldsymbol{k}_\tau/\sqrt{d_k})}{\sum_{\tau'=t-L}^{t+R} exp(\boldsymbol{q_t}\cdot\boldsymbol{k}_{\tau'}/\sqrt{d_k})} \tag{25}$$

其中L, R 代表自注意力機制所能關注的最大左右範圍，$\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v}$ 則是輸入 TRSA 層的向量，d_k 為向量 $\boldsymbol{q}, \boldsymbol{k}$ 的維度大小。$\boldsymbol{y_t}$ 為向量 $\boldsymbol{v_t}$ 的加權和，其權重為輸入向量 $\boldsymbol{q_t}$ 與前後 $L + R + 1$ 個 $\boldsymbol{k}$ 計算內積後通過 softmax 正規化所得到，之後再通過 ReLU 激活函數以及批量標準化後得到最終的輸出。同樣的 TRSA 也可以進行多頭 (Multi-head) 運算，將數個自注意力機制的結果串接起來得到輸出。

2.2.3 模型架構 (Model Architecture)

我們所使用的三種聲學模型架構如圖 4 所示，輸入的聲學特徵為 40 維的 MFCC 和 3 維的音調 (pitch) 加上 100 維的 i-vectors。首先輸入會先經過一層大小為 816，拼接窗 (splicing window) 為 (-1,0,1) 的 TDNN，接著會通過 6 層 TRSA-Transformer 架構，與一般 Transformer 架構不同的是我們使用的自注意力機制為時間限制自注意力機制。其中 TRSA 層的輸入向量 $\boldsymbol{q},\boldsymbol{k}$ 的維度大小設為 40，$\boldsymbol{v}$ 的維度大小設為 60，多頭運算的數量設為 12，時間限制的範圍 L,R 則分別設為 5 以及 2。除此之外，我們也嘗試在 TRSA 層及前饋神經網路之間加入一層 TDNN-F，以及嘗試使用兩個前饋神經網路的馬卡龍網路。其中 TDNN-F 隱藏層的維度大小為 $1536 \times 160 \times 1536$，前饋神經網路的維度大小則設為 1024。我們所使用的目標函數為 Lattice-free Maximum Mutual Information (LF-MMI) 搭配交叉熵 (Cross Entropy) 的輔助正規化訓練(Povey *et al.*, 2016)，LF-MMI 使得訓練最大化交互資訊 (Maximum Mutual Information, MMI) 時不需要事先產生詞圖 (Word Lattices)，讓訓練過程能更加快速。

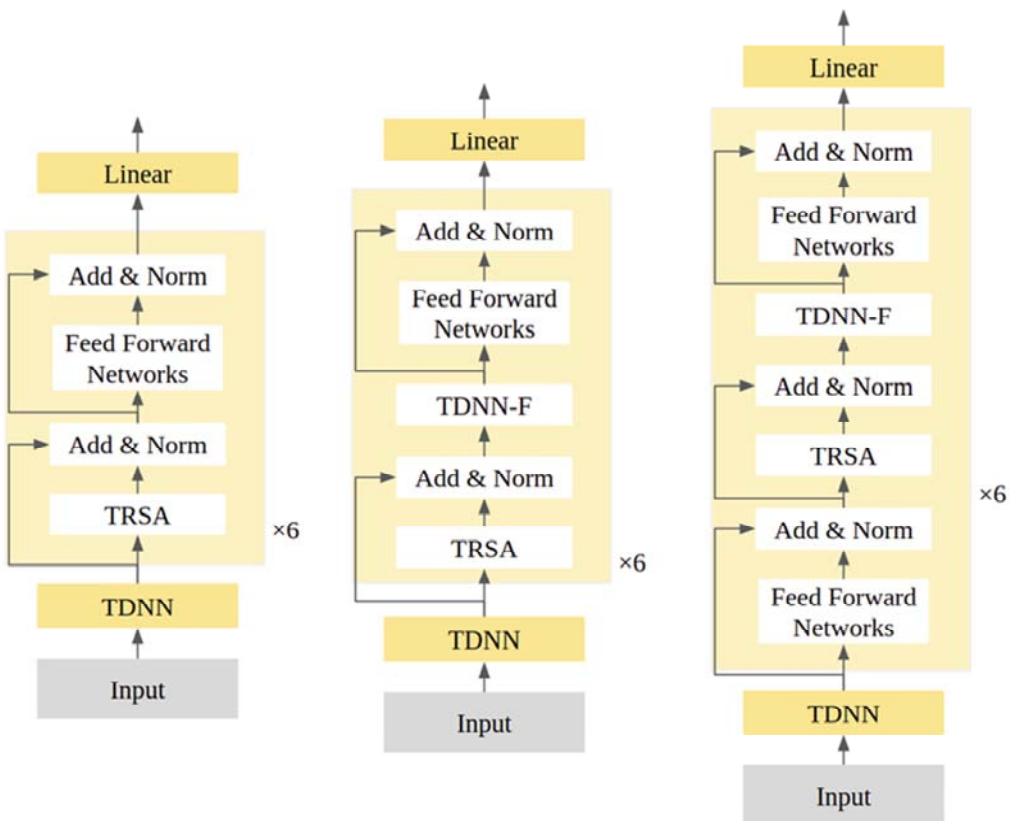

圖 4. 聲學模型架構
[Figure 4. Acoustic model architecture]

3. 實驗 (Experiments)

3.1 資料集 (Dataset)

3.1.1 Taiwanese across Taiwan (TAT) corpus

TAT-Vol1-train-lavalier 以及 TAT-Vol1-eval-lavalier 為經由競賽取得的資料集，錄音裝置為領夾式麥克風，取樣頻率為 16kHz，文本為台文漢字及台羅拼音。其中 TAT-Vol1-train-lavalier 共 80 位語者，總時長約 41.76 小時，共 23,104 筆音檔。TAT-Vol1-eval-lavalier 總時長約 4.78 小時，共 2,664 筆音檔。TAT-Vol1-train-lavalier 資料集的文本中，出現了台文漢字以及台羅拼音混用的情形。為了使同一種文本的標註一致，我們將台文漢字文本中出現非台文漢字的句子刪除，同樣的我們也將台羅拼音文本中出現非台羅拼音的句子刪除。此外我們也刪除了文本中的標點符號。

3.1.2 PTS_TW-train

PTS_TW-train 資料集為公視台語台「台灣記事簿」以及「台灣新眼界」的節目內容，文本為台文漢字，取樣頻率為 16kHz，共 95 筆音檔，總時長約 85.39 小時。TAT 以及 PTS_TW-train 資料集的數據如表 1。在 PTS_TW-train 資料集中，每句文本皆有其對應的時間標註，但是並非每筆時間標註皆精確無誤，且有部分音檔內容為中文，因此我們使用 CTC-Segmentation (Kürzinger *et al.,* 2020) 重新對齊資料集的文本及音檔。CTC-Segmentation 利用已經訓練完成且具有 CTC 輸出的模型對音檔解碼，並計算每一幀輸出欲對齊文本字元的機率，藉此取得文本的時間資訊。我們利用 CTC-Segmentation 對齊時的對數機率 (Log Probability) 選擇分數較高的音檔片段保留，最終我們根據兩種不同的對數機率閾值得到了資料集 PTS-1.5 以及 PTS-5.0，資料集的數據如表 2。

表 1. TAT 及 PTS_TW-train 資料集數據
[Table 1. Details of TAT and PTS_TW-train dataset]

資料集	音檔數	總時數	語者	文本
TAT-Vol1-train-lavalier	23,104	41.76	80	台文、台羅
TAT-Vol1-eval-lavalier	2,664	4.78	-	台文、台羅
PTS_TW-train	95	85.39	-	台文

表 2. PTS-1.5 及 PTS-5.0 資料集數據
[Table 2. Details of PTS-1.5 and PTS-5.0 dataset]

資料集	對數機率閾值	音檔數	總時數
PTS-1.5	-1.5	7,665	16.92
PTS-5.0	-5.0	23,363	52.71

3.2 實驗設置 (Experimental Setups)

我們使用兩種資料增強的方式，分別是針對音檔的變速擾動 (Speed-Perturbation) (Ko *et al.,* 2015)，額外產生語速 1.1 及 0.9 的音檔；以及針對頻譜圖進行遮蔽以及扭曲的 SpecAugment (Park *et al.,* 2019)，其中針對頻率遮蔽的大小設為 30、數量為 2，針對時間遮蔽的大小設為 40、數量為 2，針對時間扭曲的大小則設為 5。針對端到端模型，多頭注意力機制的 head 數設為 4， Conformer 的卷積模組中深度卷積層的卷積核大小設為 32，混合訓練中的超參數 α 設為 0.3，解碼階段的超參數 λ 以及 γ 分別設為 0.5 以及 0.7。我們使用 Adam 優化器 (Optimizer) (Kingma & Ba, 2014)，學習率的更新方式如 (Vaswani *et al.,* 2017) ，會在起始的 25,000 次參數更新中逐漸上升，隨後便線性下降。在解碼階段中，我們使用的語言模型為兩層 1024 個單元的長短期記憶 (Long Short-Term Memory, LSTM)，訓練語言模型的文本皆來自 TAT-Vol1-train-lavalier 及 PTS_TW-train 資料集。針對台羅拼音任務，除了使用端到端的語音辨識模型外，我們也以競賽方提供的詞典 (lexicon) 為基礎建立了 DNN-HMM 的語音辨識模型，詞典以台羅拼音的聲母以及韻母加上數字調作為音素。我們訓練了一個 monophone 的 HMM-GMM 模型以及五個 triphone 的 HMM-GMM 模型以取得訓練 DNN 時所需的狀態標籤 (State Label)，在解碼階段時則使用 tri-gram 語言模型。端到端模型使用開源套件 ESPnet (Watanabe *et al.,* 2018) 建立，並在 Nvidia GeForce GTX 1080 Ti GPU 上訓練 50 個 epochs，批量大小(Batch Size) 設為 32。DNN-HMM 模型則是使用開源套件 Kaldi (Povey *et al.,* 2011)建立，並在 Nvidia GeForce GTX 2080 Ti GPU 上訓練 12 個 epochs，批量大小設為 16。

3.3 實驗結果 (Results)

3.3.1 台文漢字實驗結果 (Hàn-jī Task Results)

針對台文漢字任務，我們使用端到端的語音辨識模型。首先，我們藉由調整 Transformer 及 Conformer 架構的層數、注意力機制的特徵維度大小以及前饋神經網路的維度大小，測試模型參數量對字元錯誤率的影響。表 3 為各種模型參數的組合及其對應的字元錯誤率，此實驗的訓練集使用 TAT-Vol1-train-lavalier，並在 TAT-Vole1-eval-lavalier 資料集上測試。由實驗結果可以看到，在相近的參數量大小時，使用 Conformer 編碼器能夠得到優於 Transformer 編碼器的結果。接著我們測試增加訓練集對模型效果的差異，我們將訓練集額外加入 PTS-1.5 以及 PTS-5.0，得到的實驗結果如表 4 所示。我們所使用的模型架構為表 3 中字元錯誤率最佳的模型，由實驗結果可以看出，加入額外的資料集進行訓練能顯著的提升模型的效能。圖 5 為各隊伍之排名及 CER。

表 3. 不同模型參數量的實驗結果
[Table 3. The results of different model parameters]

編碼器架構	Transformer	Conformer			
參數量(M)	29.5	9.9	29.6	30.7	45.4
編碼器層數	12	10	12	14	12
解碼器層數	6	4	6	4	6
注意力機制維度	256	128	256	256	256
前饋神經網路維度	2048	1024	1024	1024	2048
CER(%)	36.3	27.1	26.6	26.9	28.7

表 4. 不同訓練集的實驗結果
[Table 4. The results of the different training dataset]

訓練集	CER(%)	Final-Test CER(%)
TAT-Vol1-train-lavalier	26.6	-
TAT-Vol1-train-lavalier + PTS-1.5	21.2	48.0
TAT-Vol1-train-lavalier + PTS-5.0	19.1	43.4

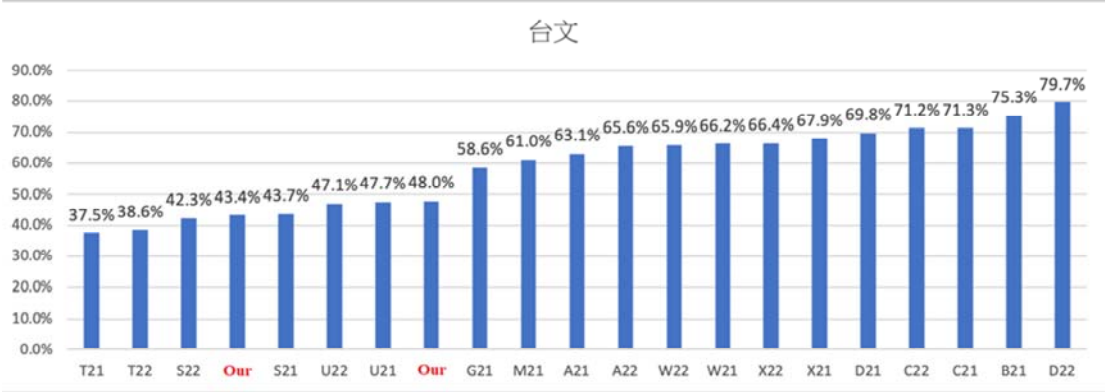

圖 5. 台文漢字競賽結果
[Figure 5. Hàn-jī challenge result]

3.3.2 台羅拼音實驗結果 (Tâi-lô Task Results)

針對端到端模型，我們比較 Transformer 編碼器以及 Conformer 編碼器的效果，以及使用不同輸出單位的差異。首先我們將每個字母及數字視為獨立的字元 (Char-based)，得到一共 37 個不同的字元；接著我們嘗試以音節為單位 (Syllable-based)，得到共 2212 個不同的音節；最後我們使用位元組對編碼 (Byte Pair Encoding, BPE) (Sennrich *et al.*, 2015)，將文本中最常出現的連續字元合併成新的 Subword，最後我們得到共 987 個不同的

Subword。表 5 為端到端模型的實驗結果，我們所使用的 Transformer 編碼器及 Conformer 編碼器的架構分別為表 3 中參數量為 29.5M 以及 29.6M 的模型。在同樣的模型架構下，使用 BPE 能得到優於其他輸出單位的結果。而在使用同樣的輸出單位時，Transformer 編碼器則取得了較佳的效果，其中以字元為單位的 Conformer 模型出現了無法收斂的情形。最終我們以使用 BPE 的 Transformer 模型在 Final-test 上取得了 25.4% 的音節錯誤率。針對 DNN-HMM，我們測試了圖 4 中三種聲學模型架構的差異，實驗結果如表 6 所示，當我們將 TDNN-F 加入 TRSA-Transformer 可以得到優於單純使用 TRSA-Transformer 的結果，而加入馬卡龍網路後又可以再進一步改進模型效能。最後我們也測試了加入 PTS 資料集的效果，由於該資料集的文本為台文漢字，因此我們利用 TAT-Vol1-train-lavalier 資料集取得共 2728 個台文漢字對應的台羅拼音，藉此取得 PTS 資料集的台羅拼音文本。實驗結果如表 7 所示，實驗中所使用的端到端模型及 DNN-HMM 模型分別為表 5 及表 6 中音節錯誤率最佳的模型，可以看出不論是端到端的語音辨識模型還是 DNN-HMM 的語音辨識模型皆隨著訓練資料量的增加而提升模型效能，其中又以端到端模型的提升效果更為明顯。圖 6 為各隊伍之排名及 SER。

表 5. 端到端模型實驗結果
[Table 5. The results of the end-to-end model]

編碼器架構	輸出單位	SER(%)	Final-Test SER(%)
Transformer	Char-based	20.8	-
	Syllable-based	20.9	-
	BPE	18.3	25.4
Conformer	Char-based	-	-
	Syllable-based	23.6	-
	BPE	19.3	-

表 6. DNN-HMM 實驗結果
[Table 6. The results of DNN-HMM]

模型架構	SER(%)	Final-Test SER(%)
TRSA-Transformer	16.3	-
TRSA-Transformer + TDNN-F	15.4	-
TRSA-Transformer + TDNN-F + 馬卡龍網路	15.3	27.1

表 7. 不同訓練集的實驗結果
[Table 7. The results of the different training dataset]

模型架構	訓練集	SER(%)
端到端語音辨識模型	TAT-Vol1-train-lavalier	18.3
	TAT-Vol1-train-lavalier + PTS-1.5	15.9
	TAT-Vol1-train-lavalier + PTS-5.0	14.3
DNN-HMM 語音辨識模型	TAT-Vol1-train-lavalier	15.3
	TAT-Vol1-train-lavalier + PTS-1.5	14.4
	TAT-Vol1-train-lavalier + PTS-5.0	13.7

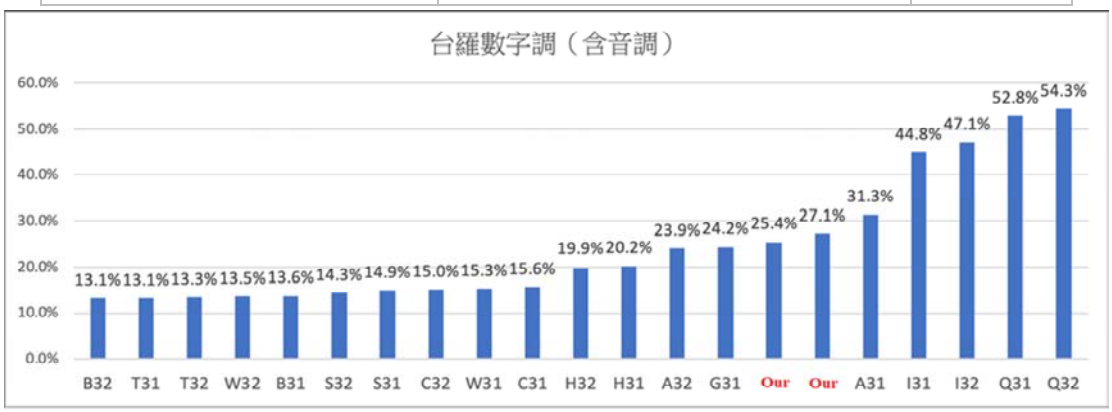

圖 6. 台羅拼音數字調競賽結果
[Figure 6. Tâi-lô challenge result]

4. 結論 (Conclusion)

我們描述了參加 FSR-2020 所使用的語音辨識系統，針對台文漢字的任務我們建立了端到端的語音辨識模型，其中編碼器使用 Conformer 架構，解碼器則為 Transformer 架構，同時也使用 CTC 損失函數共同進行訓練。我們也利用 CTC-Segmentation 對 PTS_TW-train 資料集進行前處理，以取得更多的訓練資料，最終我們取得最好的 CER 為 43.4%。針對台羅拼音任務的端到端模型則是使用 Transformer 編碼器，並且以 BPE 對文本編碼；除此之外也建立了 DNN-HMM 的語音辨識模型，聲學模型使用 TRSA-Transformer 架構結合 TDNN-F 以及馬卡龍網路。在台羅拼音任務上我們取得最好的 SER 為 25.4%。

參考文獻 (References)

Ba, J. L., Kiros, J. R. & Hinton, G. E. (2016). Layer normalization. In arXiv preprint arXiv:1607.06450

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019). Transformer-xl: Attentive language models beyond a fixed-length context. In arXiv preprint arXiv:1901.02860

Dauphin, Y. N., Fan, A., Auli, M., & Grangier, D. (2017). Language modeling with gated convolutional networks. In *Proceedings of the 34th International conference on machine learning*, 933-941. https://doi.org/10.5555/3305381.3305478

Dong, L., Xu, S. & Xu, B. (2018). Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing (ICASSP 2018)*, 5884-5888. https://doi.org/10.1109/ICASSP.2018.8462506

Graves, A., Fernández, S., Gomez, F. & Schmidhuber, J. (2006). Connectionist temporal classification: labelling unsegmented sequence data with recur-rent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, 369-376. https://doi.org/10.1145/1143844.1143891

Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., & Pang, R. (2020). Conformer: Convolution-augmented transformer for speech recognition. In arXiv preprint arXiv:2005.08100

Karita, S., Soplin, N. E. Y., Watanabe, S., Delcroix, M., Ogawa, A. & Nakatani, T. (2019). Improving Transformer-Based End-to-End Speech Recognition with Connectionist Temporal Classification and Language Model Integration. In *Proc. of Interspeech 2019*, 1408-1412. https://doi.org/10.21437/Interspeech.2019-1938

Kingma, D. P. & Ba, J. (2014). Adam: A method for stochastic optimization. In arXiv preprint arXiv:1412.6980

Ko, T., Peddinti, V., Povey, D., & Khudanpur, S. (2015). Audio augmentation for speech recognition. In *Proceedings of Interspeech 2015*, 3586-3589.

Kürzinger, L., Winkelbauer, D., Li, L., Watzel, T., & Rigoll, G. (2020). CTC-segmentation of large corpora for German end-to-end speech recognition. In *Proceedings of 22nd International Conference on Speech and Computer (SPECOM 2020)*, 267-278. https://doi.org/10.1007/978-3-030-60276-5_27

Liao, Y.-F., Chang, C.-Y., Tiun, H.-K.., Su, H.-L., Khoo, H.-L., Tsay, J. S., Tan, L.-K., Kang, P., Thiann, T.-g., Iunn, U.-G., Yang, J.-H.,…Liang, C.-N. (2020). Formosa Speech Recognition Challenge 2020 and Taiwanese across Taiwan Corpus. In *Proceedings of 23rd Conference of the Oriental COCOSDA International Committee for the Co-ordination and Standardisation of Speech Databases and Assessment Techniques (O-COCOSDA '20)*, 65-70. https://doi.org/10.1109/O-COCOSDA50338.2020.9295019

Lu, Y., Li, Z., He, D., Sun, Z., Dong, B., Qin, T., Wang, L., & Liu, T.-Y. (2019). Understanding and improving transformer from a multi-particle dynamic system point of view. In arXiv preprint arXiv:1906.02762

Park, D. S., Chan, W., Zhang, Y., Chiu, C.-C., Zoph, B., Cubuk, E. D., & Le, Q. V. (2019). Specaugment: A simple data augmentation method for automatic speech recognition. In arXiv preprint arXiv:1904.08779

Peddinti, V., Wang, Y., Povey, D. & Khudanpur, S. (2018). Low latency acoustic modeling using temporal convolution and lstms. *IEEE Signal Processing Letters*, *25*(3), 373-377. https://doi.org/10.1109/LSP.2017.2723507

Povey, D., Cheng, G., Wang, Y., Li, K., Xu, H., Yarmohammadi, M., & Khudanpur, S. (2018). Semi-orthogonal low-rank matrix factorization for deep neural networks. In *Proceedings of Interspeech 2018*, 3743-3747. https://doi.org/10.21437/Interspeech.2018-1417

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., & Vesely, K. (2011). The kaldi speech recognition toolkit. In *Proceedings of IEEE 2011 workshop on automatic speech recognition and understanding*.

Povey, D., Hadian, H., Ghahremani, P., Li, K. & Khudanpur, S. (2018). A time-restricted self-attention layer for asr. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2018)*, 5874-5878. https://doi.org/10.1109/ICASSP.2018.8462497

Povey, D., Peddinti, V., Galvez, D., Ghahremani, P., Manohar, V., Na, X., Wang, Y., & Khudanpur, S. (2016). Purely sequence-trained neural networks for asr based on lattice-free mmi. In *Proceedings of Interspeech 2016*, 2751-2755. https://doi.org/10.21437/Interspeech.2016-595

Ramachandran, P., Zoph, B. & Le, Q. V. (2017). Searching for activation functions. In arXiv preprint arXiv:1710.05941

Sennrich, R., Haddow, B., & Birch, A. (2015). Neural machine translation of rare words with subword units. In arXiv preprint arXiv:1508.07909

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. & Polosukhin, I. (2017). Attention is all you need. In arXiv preprint arXiv:1706.03762

Watanabe, S., Hori, T., Karita, S., Hayashi, T., Nishitoba, J., Unno, Y., Soplin, N. E. Y., Heymann, J., Wiesner, M., Chen, N., Renduchintala, A., & Ochiai, T. (2018). ESPnet: End-to-end speech processing toolkit. In *Proceedings of Interspeech 2018*, 2207-2211. https://doi.org/10.21437/INTERSPEECH.2018-1456