The 33rd Conference on Computational Linguistics and Speech Processing (ROCLING 2021)
Taoyuan, Taiwan, October 15-16, 2021. The Association for Computational Linguistics and Chinese Language Processing

探討領域泛化於跨裝置語者驗證系統
Discussion on domain generalization in the cross-device speaker verification system

林威廷 **Wei-Ting Lin,** 張育嘉 **Yu-Jia Zhang,** 陳嘉平 **Chia-Ping Chen**
國立中山大學資訊工程學系
National Sun Yat-sen University
Department of Computer Science and Engineering
m093040020@student.nsysu.edu.tw, m083040025@student.nsysu.edu.tw,
cpchen@mail.cse.nsysu.edu.tw
呂仲理 **Chung-Li Lu**, 詹博丞 **Bo-Cheng Chan**
中華電信研究院
Chunghwa Telecom Laboratories
chungli@cht.com.tw, cbc@cht.com.tw

摘要

本論文運用領域泛化改進跨裝置語者驗證系統的效能，我們基於一個可訓練的語者驗證系統，利用領域泛化演算法微調模型參數。首先我們使用 VoxCeleb2 資料集訓練 ECAPA-TDNN 作爲一個基準模型，接著利用 CHT-TDSV 資料集與以下領域泛化演算法來對其進行微調：DANN、CDANN、Deep CORAL。我們提出的系統在 NSYSU-TDSV 資料集中測試 10 種不同的模擬情境，包含單一裝置與多種裝置，最終於多個裝置的場景下，最佳等錯誤率從基礎模型的 18.39 下降至 8.84，成功在語者驗證系統達到跨裝置辨識的成效。

Abstract

In this paper, we use domain generalization to improve the performance of the cross-device speaker verification system. Based on a trainable speaker verification system, we use domain generalization algorithms to fine-tune the model parameters. First, we use the VoxCeleb2 dataset to train ECAPA-TDNN as a baseline model. Then, use the CHT-TDSV dataset and the following domain generalization algorithms to fine-tune it: DANN, CDNN, Deep CORAL. Our proposed system tests 10 different scenarios in the NSYSU-TDSV dataset, including a single device and multiple devices. Finally, in the scenario of multiple devices, the best equal error rate decreased from 18.39 in the baseline to 8.84. Successfully achieved cross-device identification on the speaker verification system.

關鍵字：語者驗證、領域泛化、深度神經網路

Keywords: Speaker Verification, Domain Generalization, Deep Neural Networks

1 緒論

在科技發達的現代社會當中，所見都不一定爲眞，更何況是聲音，聲音也可以經過仿冒取得利益，像是透過模仿聲音來騙取語音客服提供客户的資料，因此確認這段聲音是否爲同一個人的語者驗證技術就相當的重要。然而在不同裝置下所聽到的聲音又會有所差異，這也增加語者驗證的辨識難度。因此本研究希望解決因不同裝置之間的差異性，導致語者驗證系統辨識不佳的問題。要在跨裝置語者驗證系統中加入領域泛化 (Domain Generalization) 技術必須先訓練語者驗證系統的模型，模型在訓練過程中透過損失函數分類來學習神經網路參數，進而在推論階段能夠擷取出語者嵌入向量 (Speaker Embedding)，透過比對 Speaker Embedding 可以得到語者之間的相似度，完成語者驗證的功能。而我們再將 Speaker Embedding 當作領域泛化演算法的特徵來微調 (Fine-Tuning) 我們的模型，經過微調後我們的模型具有強健性，同個語者在不同裝置的語音可以更好的被辨識出來。

Dataset	Voxceleb2
語者數量	5,994
男性語者比例	61%
影片數量	150,480
總時長 (hours)	2,442
句子總數	1,128,246
每人平均影片數	25
每人平均句子數	185

表 1. VoxCeleb2 的詳細資訊

2 研究方法

2.1 資料集

2.1.1 VoxCeleb2

VoxCeleb2(Nagrani et al., 2020)(Chung et al., 2018)(Nagrani et al., 2017) 屬於文本無關的資料集，內容是從 Youtube 上的影片擷取聲音的片段，有名人的演講、眞人節目上的訪談、大型體育館的演說等等，因此擷取下的聲音片段會包含背景雜音，甚至於人聲、笑聲等干擾。資料集的語者範圍廣泛，涵蓋了不同年齡、職業、口音、種族。語音的採樣率爲 16kHz，單聲道，WAV 格式，使用的語言爲英文。VoxCeleb2 其他的詳細資訊如表一所示。

2.1.2 CHT-TDSV

CHT-TDSV 是中華電信研究院開發的資料集。爲文本相關的資料集，語者數爲 32 人，每位語者約有 30 到 90 筆的語音，語言爲中文，語音的採樣率爲 8kHz，單聲道，WAV 格式，語音內容爲任意 9 碼數字所組成，平均長度約 2 秒。CHT-TDSV 適用於跨裝置語者驗證，共有三種不同的裝置，分別爲麥克風、手機和市話。

2.1.3 NSYSU-TDSV

NSYSU-TDSV 是我們實驗室自行錄製的資料集。語者數爲 12 人 (9 男 3 女)，資料數共有 1,080 筆音檔，語言爲中文，語音的採樣率爲 16kHz，單聲道，WAV 格式。NSYSU-TDSV 有三種不同的裝置，分別爲麥克風 (micro)、手機 (mobile) 和市話 (office)，由這三個裝置分別作爲註冊裝置及測試裝置可以分爲 9 種情況，測試配對可以分爲註冊及測試都是同一裝置 (micro、mobile、office)，以及底線前爲註冊裝置和底線後爲測試裝置 (micro_mobile、micro_office、mobile_micro、mobile_office、office_micro、office_mobile)，最後還有以上 9 種情況全部合在一起 (all)，共十種的測試情境。

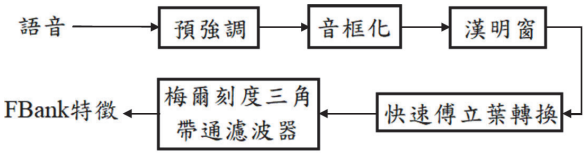

圖 1. FBank 聲學特徵處理流程

2.2 資料前處理

2.2.1 資料增強

在進行深度學習的訓練時，擁有大量的標記資料能使訓練的效果更好，也能確保訓練時不會發生過度擬合 (Over-Fitting) 的問題，因此我們利用資料增強的方法來增加我們訓練資料的數量及多樣性。我們採用兩種方法來進行資料增強，第一種是加入 MUSAN 語料庫 (Snyder et al., 2015) 的噪音，MUSAN 語料庫包含了演說 (Speech)、音樂 (Music) 與噪音 (Noise) 三個部分，演說部分的內容爲朗讀書本章節的內容或是美國政府部門的演講；音樂部分包含古典樂和現代流行樂；噪音部分包括技術性噪音 (撥號聲、傳眞機噪音等) 和環境噪音 (雷聲、雨聲、動物噪音等)，但不包括明顯可辨識說話內容的人聲。第二種爲利用房間脈衝響應 (Room Impulse Response) 加入迴響 (Reverberation)，房間脈衝響應是在房間內發出週期性的脈衝音 (Impulse Sound)，收集聲波經過房間內物體、牆面的反射所產生的迴響。

2.2.2 聲學特徵提取

在分析一段語音時，我們通常會將多個取樣點集合成一個單位，稱爲音框 (frame)，接著再從音框內提取聲學特徵作爲神經網路的輸入，這樣可以使我們的訓練更有效率。而我們採用濾波器組 (Filter bank, FBank) 作爲聲學特徵。FBank 的聲學特徵處理流程如圖一，首先語音訊號先經過預強調 (Pre-emphasis) 來對高頻的部分進行加重，使訊號的頻譜變得相對平坦，另外也是爲了補償語音訊號受到人類發音系統所限制的高頻部分。接下來將多個取樣點合成音框，再將音框代入漢明窗 (Hamming window) 函數來消除音框與音框之間可能造成的訊號不連續性。下一步是提取聲音訊號在時域上的特性，所以利用快速傅立葉轉換 (Fast Fourier Transfrom, FFT) 將其轉爲能量分布來觀察，不同的能量分布代表著不同的語音特性。再來將得到的頻譜乘上多組三角帶通濾波器來對頻譜平滑化，這樣可以使輸出的聲學特徵不受輸入語音的語調不同而有所影響。經過這些步驟後即完成 FBank 的聲學特徵提取。

2.3 ECAPA-TDNN

2.3.1 模型架構

ECAPA-TDNN(Desplanques et al., 2020)(Thienpondt et al., 2020) 是 VoxSRC-20 比賽第一名的模型，是基於時延神經網路 (TDNN)(Peddinti et al., 2015) 改進而成，我們所使用的 ECAPA-TDNN 模型 (Thienpondt et al., 2020) 架構如圖二。參數 T 代表輸入音框數、C 爲卷積通道數、k 爲卷積核大小、d 爲擴張率 (dilation rate)、S 爲語者數量，在我們的系統中，T 固定爲 200 個 frame，C 爲 2048，S 爲 5994。模型的輸入爲 80 維的特徵向量乘上 T。神經網路的第一層是 Conv1D+ReLU+BN。接下來會有 N 層的 1-D Squeeze-Excitation Res2Block(SE-Res2Block)，在 (Desplanques et al., 2020) 中只有三個 SE-Res2Block，而在 (Thienpondt et al., 2020) 中總共有四層的 SE-Res2Block，每層 SE-Res2Block 皆採用不同的擴張率，分別爲 2、3、4、5。下一層是 Conv1D+ReLU，這一層的作用爲多層特徵整合 (Multi-layer feature aggregation and summation)，將上一部分中不同擴張率的 SE-Res2Block 的輸出結合起來。接下來是 Attentive Statistical Pooling 層，計算加權平均值和加權標準差，將聚合之後的輸出進行池化。再來是一個全連接層加上 BatchNorm1d 層，用以將特徵做線性轉換得到 192 維的 embedding。最後一層是 AAM-Softmax(Deng et al., 2019)，將 192 維的 embedding 進行分類，輸出的個數等同於語者的數量 (5994)。

2.3.2 SE-Res2Block

ECAPA-TDNN 模型中最重要的部分就是 SE-Res2Block，SE-Res2Block 的架構如圖三，其實就是將 SE-Block(Hu et al., 2018) 加到 Res2Block 模組 (Gao et al., 2019) 的末端，並且將原先運用在 Res2Block 當中的 2 維卷積，改成適合語音特徵運算之具有擴張率的 1 維卷積。

SE 代表著壓縮 (Squeeze) 和激勵 (Excitation)。壓縮部分的計算方式爲公式 (1)，是針對長度 T 進行全域性平均池化 (global average pool)。h_t 代表一個 frame 的 feature map，維度爲 CxL，C 爲 channel 數，L 爲特徵幀數。經過壓縮，輸出 z 的維度成爲 Cx1。激勵部分先對每個通道的重要性進行學習，計算方式爲公式 (2)，W_1 爲 RxC 的向量，b_1 爲 Rx1 的向量，所以 $W_1 z + b_1$ 輸出一個 Rx1 的向量，其中 R 代表縮放的比例。接著 f 是一個

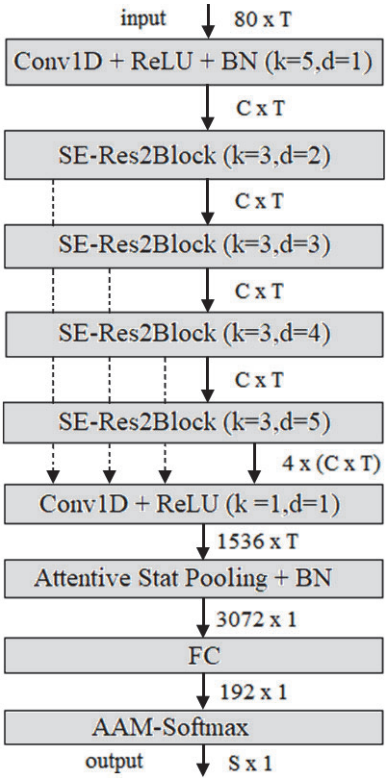

圖 2. ECAPA-TDNN 模型架構

非線性函數，在此我們使用 ReLU 函數。之後 W_2 是 CxR 的向量，b_2 爲 Cx1 的向量，所以 $W_2 f(W_1 z + b1) + b_2$ 輸出一個 Cx1 的向量。最後經過 σ 函數 (sigmoid 函數) 輸出 s，s 代表著經過全連接層、非線性層所學習到的 feature map 的權重，維度爲 Cx1。接著針對 h_t 中的每個 channel 乘上對應的權重，也就是對 h_t 做 channel-wise multiplication，如公式 (3) 所示，$s_c h_c$ 代表 s 中第 c 個 channel 和 feature map 中的第 c 個 channel 相乘，$\tilde{h_c}$ 爲第 c 個 channel 更新後的 feature map，在經過 SE 之後的 feature map 維度仍爲 CxL。

$$z = \frac{1}{T}\sum_{t}^{T} h_t \tag{1}$$

$$s = \sigma(W_2 f(W_1 z + b_1) + b_2) \tag{2}$$

$$\tilde{h_c} = s_c h_c \tag{3}$$

Res2Block 的架構如圖四，首先將特徵分爲 x1、x2、x3、x4 四組 (可以將特徵分爲任意組，這裡以四組爲例)，x1 不做任何動作即傳遞下去給 y1，而 x2 經過一組卷積大小爲 3 之卷積層提取特徵傳遞給 y2 和當作 x3 卷積層的輸入，x3 則將前一組 (x2) 的輸出和 x3 自己的輸入經過卷積層輸出給 y3 和當作 x4 卷

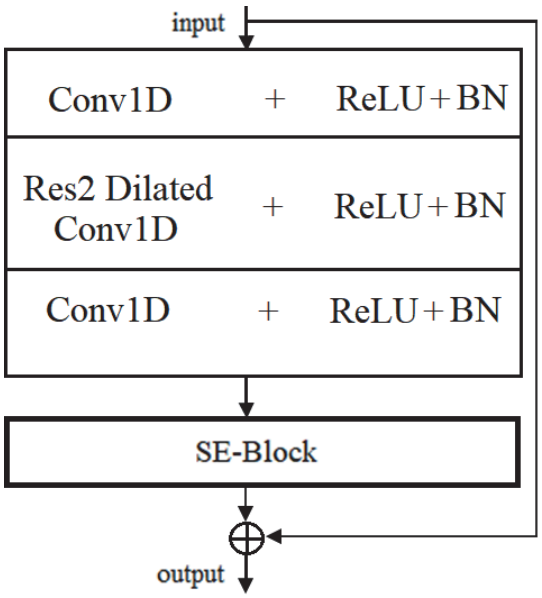

圖 3. SE-Res2Block 架構

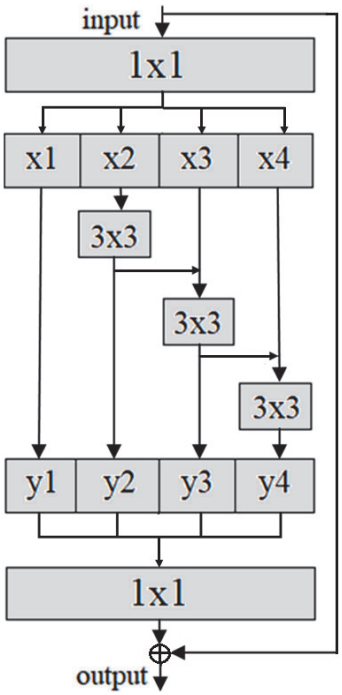

圖 4. Res2Block 架構

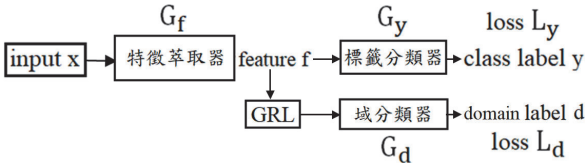

圖 5. DANN 架構

積層的輸入，x4 則將前一組 (x3) 的輸出和 x4 的輸入經過卷積層輸出給 y4，經過這些步驟之後將每一組的輸出 y1、y2、y3、y4 連接起來放進 1x1 的卷積層來將收集到的特徵整合。

2.4 領域泛化

訓練好的 ECAPA-TDNN 模型可以去分辨兩段語音是否爲同一個語者所說，但語音在錄製時會受錄製的裝置不同而導致辨識效果有所影響。我們將 ECAPA-TDNN 作爲 pre-trained model 並用領域泛化方法以 CHT-TDSV 當作訓練集進行微調來強化跨裝置語者辨識的效果，我們使用了三種不同的領域泛化方法來進行實驗，分別爲 DANN、CDANN、Deep CORAL。

2.4.1 DANN

DANN(Ganin and Lempitsky, 2015) 的原理是運用生成對抗網路 (Generative Adversarial Networks) 中對抗的概念加上深度學習技術來達到領域泛化的效果，DANN 的架構如圖五所示，主要由特徵萃取器 (feature extractor)、標籤分類器 (label predictor)、域分類器 (domain classifier) 再加上 gradient reversal layer(GRL) 所組成。DANN 架構的流程如以下說明，首先在前向傳播時輸入 x 經過特徵萃取器萃取出特徵 f，接著特徵 f 作爲標籤分類器的輸入對特徵做分類，輸出 class label y，並計算一個 loss L_y；特徵 f 作爲域分類器的輸入對 domain 進行分類，輸出 domain label d，並計算一個 loss L_d。接下來進行反向傳播，其中 θ_f、θ_y、θ_d 分別代表特徵萃取器、標籤分類器和域分類器的參數，在 L_y 的反向傳播中希望能最小化分類損失 L_y 來確保特徵 f 的判別性及分類的準確度，而在 L_d 的反向傳播過程中也希望能最小化損失 L_d，再透過 GRL 將參數乘以一個 $-\lambda$ 來反轉梯度，這樣做的目的是希望能讓域分類器不能區分源域 (source domain) 和目標域 (target domain)，從而達到領域泛化的效果。

綜合上述說明，DANN 要達到的目標是希望能最小化標籤分類器的損失 L_y，並尋找能使域分類器的損失 L_d 最大化的參數 θ_f 和能使域分類器損失 L_d 最小化的參數 θ_d，我們以公式 (4) 來表示，$\hat{\theta}_f$、$\hat{\theta}_y$、$\hat{\theta}_d$ 代表著我們要尋求的最佳解。E 函數的定義如公式 (5)，計算所有樣本在反向傳播時 loss 的總和，G_f、G_y、G_d 分別代表特徵萃取器、標籤分類器和域分類器，N 代表所有輸入的數目，x_i 表示第 i 筆輸入，y_i 表示第 i 筆輸入的 label，d_i 代表第 i 筆輸入的域標籤，0 則表示該域標籤爲源域。而 E 函數可以整理爲公式 (5) 中的第二行，L_y^i 和 L_d^i 爲在第 i 個輸入的 L_y 和

L_d。

$$\begin{aligned}\left(\hat{\theta}_f, \hat{\theta}_y\right) &= arg \min_{\theta_f, \theta_y} E\left(\theta_f, \theta_y, \hat{\theta}_d\right) \\ \hat{\theta}_d &= arg \max_{\theta_d} E\left(\hat{\theta}_f, \hat{\theta}_y, \theta_d\right)\end{aligned} \tag{4}$$

$$\begin{aligned}E\left(\theta_f, \theta_y, \theta_d\right) = & \sum_{\substack{i=1..N \\ d_i=0}} L_y\left(G_y\left(G_f\left(x_i; \theta_f\right); \theta_y\right), y_i\right) - \\ & \lambda \sum_{i=1..N} L_d\left(G_d\left(G_f\left(x_i; \theta_f\right); \theta_d\right), y_i\right) \\ = & \sum_{\substack{i=1..N \\ d_i=0}} L_y^i\left(\theta_f, \theta_y\right) - \lambda \sum_{i=1..N} L_d^i\left(\theta_f, \theta_d\right)\end{aligned} \tag{5}$$

而公式 (4) 所追求的最佳解可透過公式 (6)-(8) 的梯度更新來尋求，參數 μ 爲學習率，公式 (6) 中的 $\frac{\partial L_d^i}{\partial \theta_f}$ 乘上 $-\lambda$ 就是參數在反向傳播時反轉梯度。

$$\theta_f \leftarrow \theta_f - \mu\left(\frac{\partial L_y^i}{\partial \theta_f} + (-\lambda)\frac{\partial L_d^i}{\partial \theta_f}\right) \tag{6}$$

$$\theta_y \leftarrow \theta_y - \mu\frac{\partial L_y^i}{\partial \theta_y} \tag{7}$$

$$\theta_d \leftarrow \theta_d - \mu\frac{\partial L_d^i}{\partial \theta_d} \tag{8}$$

我們可以將 ECAPA-TDNN 模型作爲 DANN 架構的特徵萃取器和標籤分類器，特徵萃取器萃取出來的特徵等同於 ECAPA-TDNN 模型中 FC 層所輸出的 192 維的 embedding，標籤分類器所輸出的 class label 等同於 AAM-Softmax 層的輸出，因此我們可以將 ECAPA-TDNN 模型改爲圖六的架構來將兩者結合。最終 loss L_{totalD} 的計算方式可由公式 (6) 的後半部份改寫成公式 (9)，$L_{speaker}$ 和 L_{domain} 分別爲 ECAPA-TDNN 模型和域分類器的輸出。

$$L_{totalD} = L_{speaker} + (-\lambda) L_{domain} \tag{9}$$

2.4.2 CDANN

CDANN(Li et al., 2018) 爲 DANN 的一種變化，CDANN 將原本 DANN 中的域分類器加上 Prior-Normalized 成爲類先驗歸一化域分類器 (class prior-normalized domain network)，另外再加入類條件域分類器 (class-conditional domain network)，CDANN 的架構如圖七，其中 L 爲類別數，L_y、L_{norm}、L_{con} 分別爲標籤分類器、類先驗歸一化域分類器、類條件域分類器的損失，θ_f、θ_c、θ_p、θ_d 則代表在特徵萃取器、標籤分類器、類先驗歸一化域分類器和類條件域分類器神經網路上的參數。加入 class prior-based normalization 的目的爲減少各個 domain 之間 label 的分佈不一致所帶來的負面影響，而類條件域分類器的作用爲針對每個類別的資料來區分 domain。

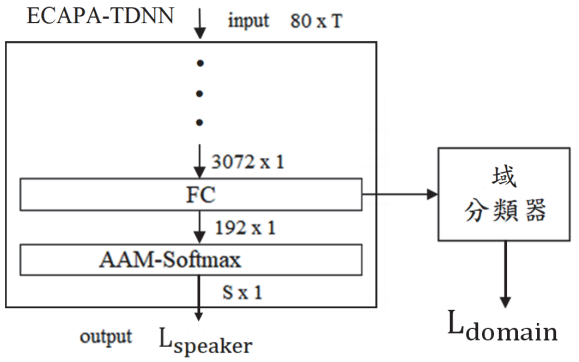

圖 6. ECAPA-TDNN + DANN 架構

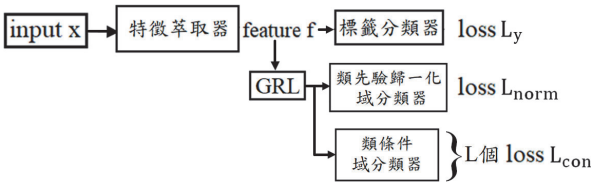

圖 7. CDANN 架構

CDANN 的訓練目標是希望能最小化標籤分類器的損失 L_y，並尋找能使類先驗歸一化域分類器和類條件域分類器的損失 L_{norm} 和 L_{con} 最大化的參數 θ_f，及尋找分別能使類先驗歸一化域分類器和類條件域分類器的損失 L_{norm} 和 L_{con} 最小化的參數 θ_p 和 θ_d，我們以公式 (10) 來表示，θ_f^*、$\left\{\theta_d^{*j}\right\}_{j=1}^{L}$、$\theta_p^*$、$\theta_c^*$ 爲我們所要尋求的最佳解，L 代表類別數，θ_d^j 代表在類別 j 中的 θ_d 參數，R 函數的定義如公式 (11)，是負責計算在反向傳播時 loss 的總和。

$$\begin{aligned}\left(\theta_f^*, \theta_c^*\right) &= \arg\min_{\theta_f, \theta_c} R\left(\theta_f, \left\{\theta_d^j\right\}_{j=1}^{L}, \theta_p, \theta_c\right) \\ \left(\left\{\theta_d^{*j}\right\}_{j=1}^{L}, \theta_p^*\right) &= \arg\max_{\left\{\theta_d^j\right\}_{j=1}^{L}, \theta_p} R\left(\theta_f, \left\{\theta_d^j\right\}_{j=1}^{L}, \theta_p, \theta_c\right)\end{aligned} \tag{10}$$

$$R\left(\theta_f, \left\{\theta_d^j\right\}_{j=1}^{L}, \theta_p, \theta_c\right) = L_y\left(\theta_f, \theta_c\right) - \lambda\left(\sum_{j=1}^{L} L_{con}\left(\theta_f, \theta_d^j\right) + L_{norm}\left(\theta_f, \theta_p\right)\right) \quad (11)$$

公式 (11) 所要尋找的最佳解可由公式 (12)-(15) 的梯度更新來尋求，參數 i 爲輸入的索引值，參數 μ 爲學習率，公式 (12)、(14)、(15) 中的 $-\lambda$ 係數使參數在梯度更新時反轉梯度。

$$\theta_f^{i+1} = \theta_f^i - \mu\left[\frac{\partial L_y^i}{\partial \theta_f} - \lambda\left(\sum_{j=1}^{L} \frac{\partial L_{con}^i\left(\theta_f, \theta_d^j\right)}{\partial \theta_f} + \frac{\partial L_{norm}^i}{\partial \theta_f}\right)\right] \quad (12)$$

$$\theta_c^{i+1} = \theta_c^i - \mu\frac{\partial L_y^i}{\partial \theta_c} \quad (13)$$

$$\left(\theta_d^j\right)^{i+1} = \left(\theta_d^j\right)^i - \mu\left(-\lambda\right)\frac{\partial L_{con}^i\left(\theta_f, \theta_d^j\right)}{\partial \theta_d^j} \quad (14)$$

$$\theta_p^{i+1} = \theta_p^i - \mu\left(-\lambda\right)\frac{\partial L_{norm}^i}{\partial \theta_p} \quad (15)$$

我們可以將 ECAPA-TDNN 模型作爲 CDANN 架構的特徵萃取器和標籤分類器，特徵萃取器萃取出來的特徵等同於 ECAPA-TDNN 模型中 FC 層所輸出的 192 維的 embedding，標籤分類器的輸出等同於 AAM-Softmax 層的輸出，因此我們可以將 ECAPA-TDNN 模型改爲圖八的架構來將兩者結合，其中 L 爲類別數。最終 loss L_{totalC} 的計算方式可由公式 (12) 的後半部份改寫成公式 (16)，$L_{speaker}$、L_{norm} 和 L_{con} 分別爲 ECAPA-TDNN 模型、類先驗歸一化域分類器和類條件域分類器的輸出。

$$L_{totalC} = L_{speaker} + \left(-\lambda\right)\left(L_{norm} + \sum_{j=1}^{L} L_{con}\right) \quad (16)$$

2.4.3 Deep CORAL

Deep CORAL(Sun and Saenko, 2016) 的架構如圖九，我們所使用的 Network 就是 ECAPA-TDNN，所以 Deep CORAL 架構的上半部分爲 ECAPA-TDNN 模型的實現，而下半部分所要達到的目標爲最小化 CORAL loss，即是最小化裝置和裝置的特徵之間共變異數的差異。目的是希望能精確的將語者分類之外，又能使不同裝置的輸出分佈更爲類似。

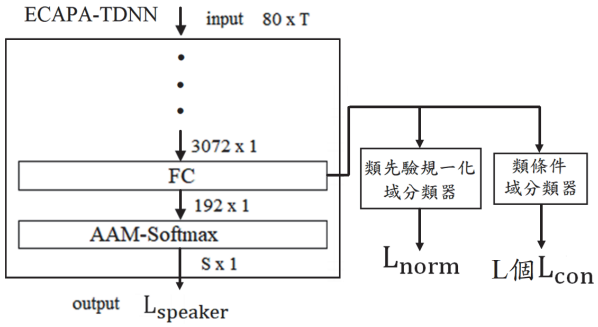

圖 8. ECAPA-TDNN + CDANN 架構

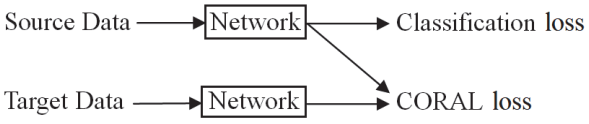

圖 9. Deep CORAL 架構

Deep CORAL 中的 CORAL loss 定義爲源域和目標域特徵的共變異數之間的距離，計算方式如公式 (17)，d 可以理解爲神經網路最後一層的輸出個數，$\|\cdot\|_F^2$ 表示 Frobenius 範數，C_S 和 C_T 分別爲源域和目標域的共變異數矩陣，C_S 和 C_T 的定義如公式 (18) 所示，減號後面的項可以理解爲平均值，D_S 爲源域的資料，D_T 爲目標域的資料，n_S 和 n_T 分別代表源域和目標域的資料個數，$\mathbf{1}$ 則是所有元素皆爲 1 的一個列向量。

$$l_{CORAL} = \frac{1}{4d^2}\left\|C_S - C_T\right\|_F^2 \quad (17)$$

$$C_S = \frac{1}{n_S - 1}\left(D_S^\tau D_S - \frac{1}{n_S}\left(\mathbf{1}^\tau D_S\right)^\tau\left(\mathbf{1}^\tau D_S\right)\right)$$
$$C_T = \frac{1}{n_T - 1}\left(D_T^\tau D_T - \frac{1}{n_T}\left(\mathbf{1}^\tau D_T\right)^\tau\left(\mathbf{1}^\tau D_T\right)\right) \quad (18)$$

最終 loss $L_{totalDC}$ 的計算方式爲公式 (19)，$L_{speaker}$ 和 L_{CORAL} 分別爲 Classification loss 和 CORAL loss，t 爲超參數，代表 CORAL loss 神經網路的層數，γ 是用來平衡 classification loss 和 CORAL loss 的超參數。

$$L_{totalDC} = L_{speaker} + \sum_{i=1}^{t}\gamma_i L_{CORAL} \quad (19)$$

3 實驗設置

3.1 實驗流程

我們系統的實驗流程如圖十所示。首先先將所有資料集 (VoxCeleb2、CHT-TDSV、

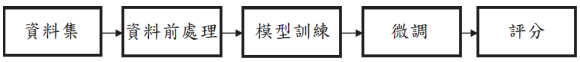

圖 10. 實驗流程圖

Test pair	Baseline	DANN	CDANN	Deep CORAL
all	18.39	8.84	13.19	9.44
micro	1.73	2.08	2.08	1.65
micro_mobile	13.25	6.07	9.17	5.30
micro_office	14.74	10.40	18.56	9.84
mobile	2.66	3.27	4.30	4.20
mobile_micro	12.87	6.62	10.60	6.21
mobile_office	11.74	8.81	15.21	8.38
office	1.56	2.60	4.16	2.60
office_micro	14.34	11.11	17.80	10.98
office_mobile	11.74	7.95	16.33	7.57

表 2. 實驗結果

NSYSU-TDSV) 經過資料前處理。之後以 VoxCeleb2 作爲訓練集訓練 ECAPA-TDNN 模型，訓練模型所使用的 batch size 爲 64，學習率爲 10^{-3}，學習率衰減爲 0.95。我們以訓練完成的 ECAPA-TDNN 模型當作 pre-trained model，使用領域泛化演算法並以 CHT-TDSV 作爲訓練集來進行微調，微調部分所使用的 batch size 爲 32，學習率爲 10^{-5}，學習率衰減爲 1。最後使用 NSYSU-TDSV 來進行測試，我們以 NSYSU-TDSV 三種不同裝置於註冊與測試之 10 種情況分別進行測試。

3.2 評分標準

我們以等錯誤率 (EER) 來當作實驗的評估標準。錯誤拒絕 (false rejection, FR) 率和錯誤接受（false acceptance, FA）率分別由公式 (20) 和公式 (21) 表示，其中參數 θ 代表接受或是拒絕的閥值，s 代表著第一筆語音 y_1 和第二筆語音 y_2 假設的相似性得分。如果 s 高於閥值表示兩段語音爲同一個語者，低於閥值則表示兩段語音爲不同的語者，透過調整閥值可以使錯誤拒絕率和錯誤接受率相等，此時的錯誤拒絕率或是錯誤接受率就是 EER。

$$P_{FR}(\theta) = P(s < \theta \mid y1 = y2) \qquad (20)$$

$$P_{FA}(\theta) = P(s > \theta \mid y1 \neq y2) \qquad (21)$$

4 實驗結果

實驗結果如表二所示，Baseline 爲不加任何領域泛化方法的 ECAPA-TDNN 模型，而 DANN、CDANN、Deep CORAL 爲各領域泛化方法之結果。

4.1 Baseline

Baseline 在三個裝置互相干擾之下有相比其他實驗方法還高的 EER，在只有兩個裝置組合的情況下也有同樣的結果，從這樣的結果可以看出跨裝置語者驗證因裝置差異導致效果變差的現象。而在相同裝置的情況下，反而是領域泛化演算法有較高的 EER，代表領域泛化演算法可能爲了要消除提取的特徵中不同裝置帶來的影響，而丟失了某些區分語者的特徵。

4.2 DANN

DANN 在三個裝置的測試得到所有演算法中最好的 EER，在其他的測試上也有良好的表現。DANN 簡單的透過反向傳播時的梯度反轉來將不同裝置的特徵分佈對齊，使裝置間的差異縮小，計算出來分數的分佈較爲接近，更能找到較好的閥值來區分語者。

4.3 CDANN

CDANN 的實驗結果爲三個領域泛化演算法中最差的。CDANN 新加入的類條件裝置分類器是想透過語者來區分不同的裝置，但語者的數量有 32 個，而且每個語者在不同裝置的特徵分佈的變換理論上是類似的，因此想透過語者來區分不同的裝置並沒有達到效果，甚至可能會讓模型的學習變得很雜亂。

4.4 Deep CORAL

在三個裝置的實驗下，Deep CORAL 的 EER 較 DANN 來的差，但在兩個裝置組合的實驗下，Deep CORAL 幾乎都表現得比 DANN 還要好。Deep CORAL 的方法爲最小化兩個域特徵的共變異數之間的距離，因此在裝置越多的情況下，不同域的特徵的共變異數會越難達到收斂，這可能是 Deep CORAL 在兩個裝置組合的實驗中較其他方法表現更好的原因。

5 結論

本論文實驗了在語者驗證系統上加入領域泛化演算法來達到跨裝置的成效，在多種領域泛化演算法的實驗下皆比原本的系統獲得大幅度的進步。不過當裝置越來越多變時，不同裝置所錄製的語音有可能差異更大，讓系統誤以爲是不同的語者，能否維持強健性是個問題，因此未來先朝實驗更多不同的裝置且達到一樣的成效爲目標。另外在領域泛化演算法上，近年來有很多應用於圖像上的領域泛化演算法，例如 Self-supervised Contrastive Regularization(Kim et al., 2021)、Smoothed-AND mask(Shahtalebi et al., 2021) 等，能否將這些演算法應用在語音領域中並套用在我們的系統上也是一個重要的改進方向。

References

Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. 2018. Voxceleb2: Deep speaker recognition. *arXiv preprint arXiv:1806.05622*.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. 2019. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699.

Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. 2020. Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. *arXiv preprint arXiv:2005.07143*.

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR.

Shanghua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip HS Torr. 2019. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*.

Jie Hu, Li Shen, and Gang Sun. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141.

Daehee Kim, Seunghyun Park, Jinkyu Kim, and Jaekoo Lee. 2021. Selfreg: Self-supervised contrastive regularization for domain generalization. *arXiv preprint arXiv:2104.09841*.

Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. 2018. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 624–639.

Arsha Nagrani, Joon Son Chung, Weidi Xie, and Andrew Zisserman. 2020. Voxceleb: Large-scale speaker verification in the wild. *Computer Speech & Language*, 60:101027.

Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. 2017. Voxceleb: a large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612*.

Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. 2015. A time delay neural network architecture for efficient modeling of long temporal contexts. In *Sixteenth annual conference of the international speech communication association*.

Soroosh Shahtalebi, Jean-Christophe Gagnon-Audet, Touraj Laleh, Mojtaba Faramarzi, Kartik Ahuja, and Irina Rish. 2021. Sand-mask: An enhanced gradient masking strategy for the discovery of invariances in domain generalization. *arXiv preprint arXiv:2106.02266*.

David Snyder, Guoguo Chen, and Daniel Povey. 2015. Musan: A music, speech, and noise corpus. *arXiv preprint arXiv:1510.08484*.

Baochen Sun and Kate Saenko. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer.

Jenthe Thienpondt, Brecht Desplanques, and Kris Demuynck. 2020. The idlab voxceleb speaker recognition challenge 2020 system description. *arXiv preprint arXiv:2010.12468*.